

[illegible]

[illegible]

(34) 3014
(35) 3061

MTHSSAB ALOG - Table for ALOG routines
MTHSSAB ATAN - Table for ATAN routines

```
0000 1      .TITLE  VMSSVECTOR - Define entry vectors for VMSRTL
0000 2      .IDENT  /4-003/                               ; File: VMSVECTOR.MAR  Edit: MDL4003
0000 3
0000 4      :
0000 5      :*****
0000 6      :*
0000 7      :*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8      :*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9      :*  ALL RIGHTS RESERVED.
0000 10     :*
0000 11     :*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12     :*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13     :*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14     :*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15     :*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16     :*  TRANSFERRED.
0000 17     :*
0000 18     :*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19     :*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20     :*  CORPORATION.
0000 21     :*
0000 22     :*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23     :*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24     :*
0000 25     :*
0000 26     :*****
0000 27     :
0000 28     :
0000 29     : FACILITY: VAX/VMS Run-Time Library
0000 30     :++
0000 31     : ABSTRACT:
0000 32     : This module contains the entry vector for the shareable image
0000 33     : VMSRTL.EXE. VMSRTL is now only a "stub" that references procedures
0000 34     : in LIBRTL, MTHRTL, BASRTL, COBRTL and FORRTL.
0000 35     :
0000 36     :--
0000 37     :
0000 38     : VERSION: 1
0000 39     :
0000 40     : Revision History:
0000 41     :
0000 42     :*****
0000 43     :*
0000 44     :*  WARNING!!!
0000 45     :*
0000 46     :*  The order or contents of the VMSRTL vector must never change!
0000 47     :*
0000 48     :*****
0000 49     :
0000 50     : 4-001 - Modified from ALLGBL.MAR to only produce vector declarations.
0000 51     :          SBL 11-May-1983
0000 52     : 4-002 - Add MTH$AB ALOG and MTH$AB ATAN table copies to end. SBL 20-May-1983
0000 53     : 4-003 - Add OLDENTRY macro for obsolete entry points. MDL 26-Sep-1983
0000 54     :--
```

```
0000 56 :+
0000 57 : NOTE: This module contains many comments which are now of only historical
0000 58 : significance. The image VMSRTL mostly consists of vectored entry
0000 59 : points that refer to procedures in other shareable images. However,
0000 60 : a few data tables that were in VMSRTL remain since they cannot be
0000 61 : revectored.
0000 62 :-
0000 63
0000 64 :+
0000 65 : Define macro MAC to generate vector entries.
0000 66 :
0000 67 : call: MAC      VEC_TYPE, VEC_AREA, SYMBOL, MASK
0000 68 :
0000 69 : where VEC_TYPE is:  CALL      - call entry point transfer vector
0000 70 :                   JSB       - JSB entry point transfer vector
0000 71 :                   NOVECT    - do not have a transfer vector
0000 72 :                   SYM       - this is a symbol, not an entry point
0000 73 :                   DATA     - this is data, kept in the vector
0000 74 :                   FUTURE    - this is a proposed entry point, not yet
0000 75 :                               implemented, but space reserved.
0000 76 :       VEC_AREA is:  FOR       - FORTRAN entry points
0000 77 :                   LIB       - library entry points
0000 78 :                   MTH       - Math library entry points
0000 79 :                   STR       - String library entry points
0000 80 :                   OTS       - Language independent entry points
0000 81 :                   BAS       - BASIC-PLUS-2 entry points
0000 82 :                   COB       - COBOL
0000 83 :
0000 84 :                   Note: VEC_AREA is ignored
0000 85 :
0000 86 :       SYMBOL is:    any entry point symbol
0000 87 :       MASK is:      optional entry mask if not same as SYMBOL
0000 88 :
0000 89 :       Each entry vector is 8 bytes long and contains a 2 byte mask and
0000 90 :       a 6 byte JMP instruction (for CALLs) or
0000 91 :       a 6 byte JMP plus 2 filler bytes for JSBs.
0000 92 :
0000 93 :-
0000 94
```

```
0000 96 .MACRO MAC VEC_TYPE, VEC_AREA, SYMBOL, MASK
0000 97 .IF IDN VEC_TYPE, JSB
0000 98 $$'SYMBOL'::
0000 99 JMP G^SYMBOL ; branch to JSB routine
0000 100 .BYTE 0,0 ; fill out to 8 bytes
0000 101 .ENDC
0000 102
0000 103 .IF IDN VEC_TYPE, CALL
0000 104 $$'SYMBOL'::
0000 105 .IF B MASK
0000 106 .MASK SYMBOL
0000 107 .IFF
0000 108 .MASK MASK ; get mask from other name
0000 109 .ENDC
0000 110 JMP G^SYMBOL+2 ; branch to CALL+2 routine
0000 111 .ENDC
0000 112
0000 113 .IF IDN VEC_TYPE, FUTURE ; Reserve space for future vector?
0000 114 .BYTE 0,0,0,0,0,0,0,0 ; leave 8 bytes
0000 115 .ENDC
0000 116
0000 117 .IF IDN VEC_TYPE, DATA
0000 118 $$'SYMBOL' V::
0000 119 .ADDRESS SYMBOL-. ; from non-shared routine. Has format:
0000 120 .BLKL 1 ;
0000 121 .ADDRESS table_name-.
0000 122 .BLKL 1
0000 123 .ENDC
0000 124 .ENDM
0000 125
0000 126 .MACRO OLDENTRY SYMBOL
0000 127 $$'SYMBOL'::
0000 128 .ENDM
0000 129
00000000 130 .PSECT $VMS$VECTOR PIC,USR,CON,REL,LCL,SHR,EXE,RD,NOWRT,PAGE
0000 131 RTL$START:
0000 132
```

```
0000 134 ;+
0000 135 ; FORTTRAN compatibility routines - do not VECTOR
0000 136 ; -
0000 137
0000 138
0000 139 ; MODULE:COM$ASSIGN
0000 140 MAC NOVECT COM ASSIGN
0000 141
0000 142 ; MODULE:COM$CLOSE
0000 143 MAC NOVECT COM CLOSE
0000 144
0000 145 ; MODULE:COM$ERRSET
0000 146 MAC NOVECT COM ERRSET
0000 147
0000 148 ; MODULE:COM$ERRTST
0000 149 MAC NOVECT COM ERRTST
0000 150
0000 151 ; MODULE:COM$FDBSET
0000 152 MAC NOVECT COM FDBSET
0000 153
0000 154 ; MODULE:COM$IRAD50
0000 155 MAC NOVECT COM IRAD50
0000 156
0000 157 ; MODULE:COM$R50ASC
0000 158 MAC NOVECT COM R50ASC
0000 159
0000 160 ; MODULE:COM$RAD50
0000 161 MAC NOVECT COM RAD50
0000 162
0000 163 ; MODULE:COM$USEREX
0000 164 MAC NOVECT COM USEREX
```

```
0000 166 :+
0000 167 :+ FORTRAN entry points
0000 168 :+ Put most frequently used FORTRAN entry points together first,
0000 169 :+ ie. I/O and OPEN and CLOSE.
0000 170 :-
0000 171
0000 172 : MODULE:FOR$CLOSE
0000 173 : MAC CALL FOR FOR$CLOSE
0008 174 : MODULE:FOR$ENTRY
0008 175 : MAC CALL FOR FOR$DECODE_MF FOR$$IO_BEG
0010 176
0010 177 : MAC CALL FOR FOR$DECODE_MO FOR$$IO_BEG
0018 178 : MAC CALL FOR FOR$ENCODE_MF FOR$$IO_BEG
0020 179 : MAC CALL FOR FOR$ENCODE_MO FOR$$IO_BEG
0028 180
0028 181 : MAC CALL FOR FOR$READ_KF FOR$$IO_BEG
0030 182 : MAC CALL FOR FOR$READ_KO FOR$$IO_BEG
0038 183
0038 184 : MAC CALL FOR FOR$READ_DF FOR$$IO_BEG
0040 185 : MAC CALL FOR FOR$READ_DO FOR$$IO_BEG
0048 186 : MAC CALL FOR FOR$READ_DU FOR$$IO_BEG
0050 187 : MAC CALL FOR FOR$READ_SF FOR$$IO_BEG
0058 188 : MAC CALL FOR FOR$READ_SL FOR$$IO_BEG
0060 189
0060 190 : MAC CALL FOR FOR$READ_SO FOR$$IO_BEG
0068 191 : MAC CALL FOR FOR$READ_SU FOR$$IO_BEG
0070 192 : MAC CALL FOR FOR$WRITE_DF FOR$$IO_BEG
0078 193 : MAC CALL FOR FOR$WRITE_DO FOR$$IO_BEG
0080 194
0080 195 : MAC CALL FOR FOR$WRITE_DU FOR$$IO_BEG
0088 196 : MAC CALL FOR FOR$WRITE_SF FOR$$IO_BEG
0090 197 : MAC CALL FOR FOR$WRITE_SL FOR$$IO_BEG
0098 198 : MAC CALL FOR FOR$WRITE_SO FOR$$IO_BEG
00A0 199
00A0 200 : MAC CALL FOR FOR$WRITE_SU FOR$$IO_BEG
00A8 201
00A8 202 : MODULE:FOR$IO_END
00A8 203 : MAC CALL FOR FOR$IO_END
00B0 204
00B0 205 : MODULE:FOR$IO_ELEM
00B0 206
00B0 207 : MAC CALL FOR FOR$IO_F_R
00B8 208 : MAC CALL FOR FOR$IO_F_V
00C0 209 : MAC CALL FOR FOR$IO_D_R
00C8 210 : MAC CALL FOR FOR$IO_D_V
00D0 211
00D0 212 : MAC CALL FOR FOR$IO_L_R
00D8 213 : MAC CALL FOR FOR$IO_L_V
00E0 214 : MAC CALL FOR FOR$IO_B_R
00E8 215 : MAC CALL FOR FOR$IO_B_V
00F0 216
00F0 217 : MAC CALL FOR FOR$IO_T_DS
00F8 218
00F8 219 : MAC CALL FOR FOR$IO_W_R
0100 220 : MAC CALL FOR FOR$IO_W_V
0108 221 : MAC CALL FOR FOR$IO_G_R
0110 222 : MAC CALL FOR FOR$IO_G_V
```

```
0118 223      MAC      CALL      FOR      FOR$IO_H_R
0120 224      MAC      CALL      FOR      FOR$IO_H_V
0128 225
0128 226      MAC      CALL      FOR      FOR$IO_DC_R
0130 227      MAC      CALL      FOR      FOR$IO_GC_R
0138 228 :      by value calls at end
0138 229      MAC      CALL      FOR      FOR$IO_T_V_DS
0140 230      MAC      CALL      FOR      FOR$IO_FC_R
0148 231      MAC      CALL      FOR      FOR$IO_FC_V
0150 232      MAC      CALL      FOR      FOR$IO_LU_R
0158 233      MAC      CALL      FOR      FOR$IO_LU_V
0160 234      MAC      CALL      FOR      FOR$IO_WU_R
0168 235      MAC      CALL      FOR      FOR$IO_WU_V
0170 236      MAC      CALL      FOR      FOR$IO_X_DA
0178 237
0178 238 : MODULE:FOR$OPEN
0178 239      MAC      CALL      FOR      FOR$OPEN
0180 240
0180 241 :+
0180 242 : Rest of FOR$ entries alphabetical order
0180 243 :-
0180 244
0180 245 : MODULE:FOR$BACKSPACE
0180 246      MAC      CALL      FOR      FOR$BACKSPACE
0188 247
0188 248 : MODULE:FOR$BITOPS
0188 249      MAC      NOVECT      FOR      FOR$IMVBITS
0188 250      MAC      NOVECT      FOR      FOR$JMVBITS
0188 251      MAC      NOVECT      FOR      FOR$IIBITS
0188 252      MAC      NOVECT      FOR      FOR$JIBITS
0188 253      MAC      NOVECT      FOR      FOR$IISHFTC
0188 254      MAC      NOVECT      FOR      FOR$JISHFTC
0188 255      MAC      NOVECT      FOR      FOR$BITEST
0188 256      MAC      NOVECT      FOR      FOR$BJTEST
0188 257      MAC      NOVECT      FOR      FOR$IIBSET
0188 258      MAC      NOVECT      FOR      FOR$JIBSET
0188 259      MAC      NOVECT      FOR      FOR$IIBCLR
0188 260      MAC      NOVECT      FOR      FOR$JIBCLR
0188 261
0188 262 : MODULE:OTSS$CVTLT      ; New entry points at end
0188 263      MAC      CALL      FOR      FOR$CNV_OUT_I
0190 264      MAC      CALL      FOR      FOR$CNV_OUT_L
0198 265      MAC      CALL      FOR      FOR$CNV_OUT_O
01A0 266      MAC      CALL      FOR      FOR$CNV_OUT_Z
01A8 267
01A8 268 : MODULE FOR$CVTRT - replaces FOR$CNV_OUT
01A8 269      OLDENTRY      FOR$CNV_OUT_D
01A8 270      MAC      CALL      FOR      FOR$CVT_D_TD
01B0 271      OLDENTRY      FOR$CNV_OUT_E
01B0 272      MAC      CALL      FOR      FOR$CVT_D_TE
01B8 273      OLDENTRY      FOR$CNV_OUT_F
01B8 274      MAC      CALL      FOR      FOR$CVT_D_TF
01C0 275      OLDENTRY      FOR$CNV_OUT_G
01C0 276      MAC      CALL      FOR      FOR$CVT_D_TG
01C8 277
01C8 278 : MODULE:FOR$DATE
01C8 279      MAC      NOVECT      FOR      FOR$DATE
```

01C8	280				
01C8	281	; MODULE:FOR\$DATE_T_DS			
01C8	282	MAC NOVECT	FOR	FOR\$DATE_T_DS	
01C8	283				
01C8	284	; MODULE:FOR\$DEFINE_FILE			
01C8	285	MAC CALC	FOR	FOR\$DEF_FILE	
01D0	286	MAC CALL	FOR	FOR\$DEF_FILE_W	
01D8	287				
01D8	288	; MOUDLE FOR\$ENDFILE			
01D8	289	MAC CALL	FOR	FOR\$ENDFILE	
01E0	290				
01E0	291	; MODULE:FOR\$ENODEF			
01E0	292	MAC SYM	FOR	FOR\$K_ADJARRDIM	
01E0	293	MAC SYM	FOR	FOR\$K_ARRREFOUT	
01E0	294	MAC SYM	FOR	FOR\$K_ATTACCNON	
01E0	295	MAC SYM	FOR	FOR\$K_BACERR	
01E0	296				
01E0	297	MAC SYM	FOR	FOR\$K_CLOERR	
01E0	298	MAC SYM	FOR	FOR\$K_DECSTROVE	
01E0	299	MAC SYM	FOR	FOR\$K_DELERR	
01E0	300	MAC SYM	FOR	FOR\$K_DUPFILSPE	
01E0	301	MAC SYM	FOR	FOR\$K_ENDDURREA	
01E0	302	MAC SYM	FOR	FOR\$K_ENDFILERR	
01E0	303	MAC SYM	FOR	FOR\$K_ERRDURREA	
01E0	304	MAC SYM	FOR	FOR\$K_ERRDURWRI	
01E0	305	MAC SYM	FOR	FOR\$K_FAC_NO	
01E0	306				
01E0	307	MAC SYM	FOR	FOR\$K_FILNAMSPE	
01E0	308	MAC SYM	FOR	FOR\$K_FILNOTFOU	
01E0	309	MAC SYM	FOR	FOR\$K_FINERR	
01E0	310	MAC SYM	FOR	FOR\$K_FLOOVE	
01E0	311	MAC SYM	FOR	FOR\$K_FLOUND	
01E0	312				
01E0	313	MAC SYM	FOR	FOR\$K_FLOZERDIV	
01E0	314	MAC SYM	FOR	FOR\$K_FORVARMIS	
01E0	315	MAC SYM	FOR	FOR\$K_INCFILORG	
01E0	316	MAC SYM	FOR	FOR\$K_INCKEYCHG	
01E0	317	MAC SYM	FOR	FOR\$K_INCOPECLO	
01E0	318	MAC SYM	FOR	FOR\$K_INCRECLEN	
01E0	319	MAC SYM	FOR	FOR\$K_INCRECTYP	
01E0	320	MAC SYM	FOR	FOR\$K_INFFORLOO	
01E0	321	MAC SYM	FOR	FOR\$K_INPCONERR	
01E0	322	MAC SYM	FOR	FOR\$K_INPRECTOO	
01E0	323	MAC SYM	FOR	FOR\$K_INPSTAREQ	
01E0	324	MAC SYM	FOR	FOR\$K_INSVIRMEM	
01E0	325	MAC SYM	FOR	FOR\$K_INTOVF	
01E0	326	MAC SYM	FOR	FOR\$K_INTZERDIV	
01E0	327	MAC SYM	FOR	FOR\$K_INVARGFOR	
01E0	328	MAC SYM	FOR	FOR\$K_INVKEYSPE	
01E0	329	MAC SYM	FOR	FOR\$K_INVLOGUNI	
01E0	330	MAC SYM	FOR	FOR\$K_INVREFVAR	
01E0	331	MAC SYM	FOR	FOR\$K_KEYVALERR	
01E0	332	MAC SYM	FOR	FOR\$K_LISIO_SYN	
01E0	333	MAC SYM	FOR	FOR\$K_MAX_ERR	
01E0	334	MAC SYM	FOR	FOR\$K_MIXFILACC	
01E0	335	MAC SYM	FOR	FOR\$K_NOTFORSPE	
01E0	336	MAC SYM	FOR	FOR\$K_NO_CURREC	

```
01E0 337 MAC SYM FOR FOR$K_NO_SUCDEV
01E0 338 MAC SYM FOR FOR$K_OPEDEFREQ
01E0 339 MAC SYM FOR FOR$K_OPEFAI
01E0 340
01E0 341 MAC SYM FOR FOR$K_OUTCONERR
01E0 342 MAC SYM FOR FOR$K_OUTSTAOVE
01E0 343 MAC SYM FOR FOR$K_RECIO_OPE
01E0 344 MAC SYM FOR FOR$K_RECNUMOUT
01E0 345 MAC SYM FOR FOR$K_REWERR
01E0 346 MAC SYM FOR FOR$K_REWRITERR
01E0 347 MAC SYM FOR FOR$K_SEGRECFOR
01E0 348 MAC SYM FOR FOR$K_SPERECLOC
01E0 349 MAC SYM FOR FOR$K_SYNERRFOR
01E0 350 MAC SYM FOR FOR$K_SYNERRNAM
01E0 351 MAC SYM FOR FOR$K_TOOMANREC
01E0 352 MAC SYM FOR FOR$K_TOOMANVAL
01E0 353 MAC SYM FOR FOR$K_UNIALROPE
01E0 354 MAC SYM FOR FOR$K_UNLERR
01E0 355 MAC SYM FOR FOR$K_VFEVALERR
01E0 356 MAC SYM FOR FOR$K_WRIREALFIL
01E0 357
01E0 358 ; MODULE:FOR$ERRSNS ; See also end where FOR$$ERRSNS_SAV declared
01E0 359 MAC CALL FOR FOR$ERRSNS
01E8 360 MAC CALL FOR FOR$ERRSNS_W
01F0 361
01F0 362 ; MODULE:FOR$EXIT
01F0 363 MAC CALL FOR FOR$EXIT
01F8 364 MAC CALL FOR FOR$EXIT_W
0200 365
0200 366 ; MODULE:OTSS$CVTTR
0200 367 ; This module is a replacement
0200 368 ; for FOR$CNVIR. The old
0200 369 ; FOR$ entry point still exists.
0200 370 ; See later where other OTSS
0200 371 ; entry points are named.
0200 372
0200 373 OLDENTRY FOR$CNV_IN DEFG ; Same as next symbol.
0200 374 MAC CALL OTS -OTSS$CVT_T_D
0208 375
0208 376 ; MODULE:FOR$FIND
0208 377 MAC CALL FOR FOR$FIND
0210 378
0210 379 ; MODULE:OTSS$CVT_TI_L - replaces in part FOR$CNVII
0210 380 OLDENTRY FOR$CNV_IN_I
0210 381 MAC CALL OTS -OTSS$CVT_TI_L
0218 382
0218 383 ; MODULE:OTSS$CVT_TL_L - replaces in part FOR$CNVII
0218 384 OLDENTRY FOR$CNV_IN_L
0218 385 MAC CALL OTS -OTSS$CVT_TL_L
0220 386
0220 387 ; MODULE:OTSS$CVT_TO_L - replaces in part FOR$CNVII
0220 388 OLDENTRY FOR$CNV_IN_O
0220 389 MAC CALL OTS -OTSS$CVT_TO_L
0228 390 OLDENTRY FOR$CNV_IN_Z
0228 391 MAC CALL OTS -OTSS$CVT_TZ_L
0230 392
0230 393 ; MODULE:FOR$IDATE
```

0230	394	MAC	NOVECT	FOR	FOR\$IDATE
0230	395				
0230	396	; MODULE:FOR\$INI_DES			
0230	397	MAC	JSB	FOR	FOR\$INI_DES1_R2
0238	398	MAC	JSB	FOR	FOR\$INI_DES2_R3
0240	399	MAC	JSB	FOR	FOR\$INI_DESC_R6
0248	400				
0248	401	; MODULE:FOR\$JDATE			
0248	402	MAC	NOVECT	FOR	FOR\$JDATE
0248	403				
0248	404	; MODULE:FOR\$MSGDEF			
0248	405	MAC	SYM	FOR	FOR\$_ADJARRDIM
0248	406	MAC	SYM	FOR	FOR\$_ATTACCNON
0248	407	MAC	SYM	FOR	FOR\$_BACERR
0248	408	MAC	SYM	FOR	FOR\$_CLOERR
0248	409				
0248	410	MAC	SYM	FOR	FOR\$_DELERR
0248	411	MAC	SYM	FOR	FOR\$_DUPFILSPE
0248	412	MAC	SYM	FOR	FOR\$_ENDDURREA
0248	413	MAC	SYM	FOR	FOR\$_ENDFILERR
0248	414	MAC	SYM	FOR	FOR\$_ERRDURREA
0248	415				
0248	416	MAC	SYM	FOR	FOR\$_ERRDURWRI
0248	417	MAC	SYM	FOR	FOR\$_FILNAMSPE
0248	418	MAC	SYM	FOR	FOR\$_FILNOTFOU
0248	419	MAC	SYM	FOR	FOR\$_FINERR
0248	420	MAC	SYM	FOR	FOR\$_FORVARMIS
0248	421				
0248	422	MAC	SYM	FOR	FOR\$_INCFILORG
0248	423	MAC	SYM	FOR	FOR\$_INCKEYCHG
0248	424	MAC	SYM	FOR	FOR\$_INCOPECLO
0248	425	MAC	SYM	FOR	FOR\$_INCRECLEN
0248	426	MAC	SYM	FOR	FOR\$_INCRECTYP
0248	427	MAC	SYM	FOR	FOR\$_INFFORLOO
0248	428	MAC	SYM	FOR	FOR\$_INPCONERR
0248	429				
0248	430	MAC	SYM	FOR	FOR\$_INPRECTOO
0248	431	MAC	SYM	FOR	FOR\$_INPSTAREQ
0248	432	MAC	SYM	FOR	FOR\$_INSVIRMEM
0248	433	MAC	SYM	FOR	FOR\$_INVARGFOR
0248	434	MAC	SYM	FOR	FOR\$_INVKEYSPE
0248	435				
0248	436	MAC	SYM	FOR	FOR\$_INVLOGUNI
0248	437	MAC	SYM	FOR	FOR\$_INVREFVAR
0248	438	MAC	SYM	FOR	FOR\$_KEYVALERR
0248	439	MAC	SYM	FOR	FOR\$_LISIO_SYN
0248	440	MAC	SYM	FOR	FOR\$_MIXFILACC
0248	441				
0248	442	MAC	SYM	FOR	FOR\$_NOTFORSPE
0248	443	MAC	SYM	FOR	FOR\$_NO_CURREC
0248	444	MAC	SYM	FOR	FOR\$_NO_SUCDEV
0248	445	MAC	SYM	FOR	FOR\$_OPEDEFREQ
0248	446				
0248	447	MAC	SYM	FOR	FOR\$_OPEFAI
0248	448	MAC	SYM	FOR	FOR\$_OUTCONERR
0248	449	MAC	SYM	FOR	FOR\$_OUTSTAOVE
0248	450	MAC	SYM	FOR	FOR\$_RECIO_OPE

0248	451				
0248	452	MAC	SYM	FOR	FOR\$_RECNUMOUT
0248	453	MAC	SYM	FOR	FOR\$_REWERR
0248	454	MAC	SYM	FOR	FOR\$_REWPITERR
0248	455	MAC	SYM	FOR	FOR\$_SEGRCFOR
0248	456				
0248	457	MAC	SYM	FOR	FOR\$_SPERECLOC
0248	458	MAC	SYM	FOR	FOR\$_SYNERRFOR
0248	459	MAC	SYM	FOR	FOR\$_SYNERRNAM
0248	460	MAC	SYM	FOR	FOR\$_TOOMANREC
0248	461	MAC	SYM	FOR	FOR\$_TOOMANVAL
0248	462	MAC	SYM	FOR	FOR\$_UNIALROPE
0248	463	MAC	SYM	FOR	FOR\$_UNLERR
0248	464	MAC	SYM	FOR	FOR\$_VFEVALERR
0248	465	MAC	SYM	FOR	FOR\$_WRIREFIL
0248	466				
0248	467	; MODULE:FOR\$PAUSE			
0248	468	MAC	CALL	FOR	FOR\$PAUSE
0250	469				
0250	470	; MODULE:FOR\$RANDOM			
0250	471	MAC	NOVECT	FOR	FOR\$IRAN
0250	472	MAC	NOVECT	FOR	FOR\$RANDU
0250	473	MAC	NOVECT	FOR	FOR\$RANDU_W
0250	474				
0250	475	; MODULE:FOR\$REWIND			
0250	476	MAC	CALL	FOR	FOR\$REWIND
0258	477				
0258	478	; MODULE:FOR\$SECNDS			
0258	479	MAC	CALL	FOR	FOR\$SECNDS
0260	480				
0260	481	; MODULE:FOR\$STOP			
0260	482	MAC	CALL	FOR	FOR\$STOP
0268	483				
0268	484	; MODULE:FOR\$TIME			
0268	485	MAC	NOVECT	FOR	FOR\$TIME
0268	486				
0268	487	; MODULE:FOR\$TIME_T_DS			
0268	488	MAC	NOVECT	FOR	FOR\$TIME_T_DS
0268	489				

```
0268 491 ;+
0268 492 ; Mathematical library entry points
0268 493 ; Include the frequently used ones first (ones with JSBs)
0268 494 ; -
0268 495
0268 496 ; MODULE:MTH$ACOS (Degree entries further on)
0268 497     MAC      CALL      MTH      MTH$ACOS
0270 498     OLDENTRY MTH$ACOS_R5      ; Release 1 name
0270 499     MAC      JSB      MTH      MTH$ACOS_R4
0278 500
0278 501 ; MODULE:MTH$ALOG
0278 502     MAC      CALL      MTH      MTH$ALOG
0280 503     MAC      CALL      MTH      MTH$ALOG10
0288 504     MAC      JSB      MTH      MTH$ALOG10_R5
0290 505     MAC      JSB      MTH      MTH$ALOG_R5
0298 506
0298 507 ; MODULE:MTH$ASIN
0298 508     MAC      CALL      MTH      MTH$ASIN
02A0 509     OLDENTRY MTH$ASIN_R5      ; Release 1 name
02A0 510     MAC      JSB      MTH      MTH$ASIN_R4
02A8 511
02A8 512 ; MODULE:MTH$ATAN
02A8 513     MAC      CALL      MTH      MTH$ATAN
02B0 514     MAC      CALL      MTH      MTH$ATAN2
02B8 515     MAC      JSB      MTH      MTH$ATAN_R4
02C0 516
02C0 517 ; MODULE:MTH$DACOS
02C0 518     MAC      CALL      MTH      MTH$DACOS
02C8 519     OLDENTRY MTH$DACOS_R9      ; Release 1 name
02C8 520     MAC      JSB      MTH      MTH$DACOS_R7
02D0 521
02D0 522 ; MODULE:MTH$DASIN
02D0 523     MAC      CALL      MTH      MTH$DASIN
02D8 524     OLDENTRY MTH$DASIN_R9      ; Release 1 name
02D8 525     MAC      JSB      MTH      MTH$DASIN_R7
02E0 526
02E0 527 ; MODULE:MTH$DATAN
02E0 528     MAC      CALL      MTH      MTH$DATAN
02E8 529     MAC      CALL      MTH      MTH$DATAN2
02F0 530     MAC      JSB      MTH      MTH$DATAN_R7
02F8 531
02F8 532 ; MODULE:MTH$DEXP
02F8 533     MAC      CALL      MTH      MTH$DEXP
0300 534     OLDENTRY MTH$DEXP_R7      ; Obsolete name
0300 535     MAC      JSB      MTH      MTH$DEXP_R6
```

```
0308 537 ; MODULE:MTH$DLOG
0308 538     MAC      CALL    MTH      MTH$DLOG
0310 539     MAC      CALL    MTH      MTH$DLOG10
0318 540     MAC      JSB     MTH      MTH$DLOG10_R8
0320 541     MAC      JSB     MTH      MTH$DLOG_R8
0328 542
0328 543 ; MODULE:MTH$DSINCOS
0328 544     MAC      CALL    MTH      MTH$DCOS
0330 545     MAC      JSB     MTH      MTH$DCOS_R7
0338 546     MAC      CALL    MTH      MTH$DSIN
0340 547     MAC      JSB     MTH      MTH$DSIN_R7
0348 548
0348 549 ; MODULE:MTH$DSQRT
0348 550     MAC      CALL    MTH      MTH$DSQRT
0350 551     MAC      JSB     MTH      MTH$DSQRT_R5
0358 552
0358 553 ; MODULE:MTH$EXP
0358 554     MAC      CALL    MTH      MTH$EXP
0360 555     MAC      JSB     MTH      MTH$EXP_R4
0368 556
0368 557 ; MODULE:MTH$SINCOS
0368 558     MAC      CALL    MTH      MTH$COS
0370 559     MAC      JSB     MTH      MTH$COS_R4
0378 560     MAC      CALL    MTH      MTH$SIN
0380 561     MAC      JSB     MTH      MTH$SIN_R4
0388 562
0388 563 ; MODULE:MTH$SQRT
0388 564     MAC      CALL    MTH      MTH$SQRT
0390 565 ;      JSB to MTH$SQRT_R3 is with new entries.
0390 566
0390 567 ; MODULE:MTH$SQRT_R2      (obsolete module)
0390 568     MAC      JSB     MTH      MTH$SQRT_R2
0398 569
```

```
0398 571 :+
0398 572 : Language independent support entry points
0398 573 : Include them after frequently used math routines, since
0398 574 : they have the power routines.
0398 575 :-
0398 576
0398 577
0398 578 ; MODULE:OTSS$DIVC
0398 579     MAC      CALL      OTS      OTSS$DIVC
03A0 580
03A0 581 ; MODULE:OTSS$LINKAGE
03A0 582     MAC      SYM      OTS      OTSS$LINKAGE
03A0 583
03A0 584 ; MODULE:OTSS$MSGDEF
03A0 585     MAC      SYM      OTS      OTSS$_FATINTERR
03A0 586     MAC      SYM      OTS      OTSS$_INPCONERR
03A0 587     MAC      SYM      OTS      OTSS$_INTDATCOR
03A0 588     MAC      SYM      OTS      OTSS$_INVSTRDES
03A0 589     MAC      SYM      OTS      OTSS$_IO_CONCLO
03A0 590     MAC      SYM      OTS      OTSS$_OUTCONERR
03A0 591     MAC      SYM      OTS      OTSS$_USEFLORES
03A0 592     MAC      SYM      OTS      OTSS$_WRONUMARG
03A0 593
03A0 594 ; MODULE:OTSS$POWCJ
03A0 595     MAC      CALL      OTS      OTSS$POWCJ
03A8 596
03A8 597 ; MODULE:OTSS$POWDD
03A8 598     MAC      CALL      OTS      OTSS$POWDD
03B0 599     MAC      CALL      OTS      OTSS$POWDR
03B8 600     MAC      CALL      OTS      OTSS$POWRD
03C0 601
03C0 602 ; MODULE:OTSS$POWDJ
03C0 603     MAC      CALL      OTS      OTSS$POWDJ
03C8 604
03C8 605 ; MODULE:OTSS$POWII
03C8 606     MAC      CALL      OTS      OTSS$POWII
03D0 607
03D0 608 ; MODULE:OTSS$POWJJ
03D0 609     MAC      CALL      OTS      OTSS$POWJJ
03D8 610
03D8 611 ; MODULE:OTSS$POWRJ
03D8 612     MAC      CALL      OTS      OTSS$POWRJ
03E0 613
03E0 614 ; MODULE:OTSS$POWRR
03E0 615     MAC      CALL      OTS      OTSS$POWRR
03E8 616
03E8 617 ; MODULE:OTSS$SCOPY
03E8 618     MAC      CALL      OTS      OTSS$SCOPY_DXDX
03F0 619     MAC      JSB      OTS      OTSS$SCOPY_DXDX6
03F8 620     MAC      CALL      OTS      OTSS$SCOPY_R_DX
0400 621     MAC      JSB      OTS      OTSS$SCOPY_R_DX6
0408 622     MAC      CALL      OTS      OTSS$SGET1_DD
0410 623     MAC      JSB      OTS      OTSS$SGET1_DD_R6
0418 624     MAC      CALL      OTS      OTSS$SFREET_DD
0420 625     MAC      JSB      OTS      OTSS$SFREET_DD6
0428 626     MAC      CALL      OTS      OTSS$SFREEN_DD
0430 627     MAC      JSB      OTS      OTSS$SFREEN_DD6
```

```
0438 629 :+
0438 630 : Now define the rest of the Math entry points
0438 631 :-
0438 632 :
0438 633 ; MODULE:MTH$ABS
0438 634 MAC NOVECT MTH MTH$ABS
0438 635 MAC NOVECT MTH MTH$DABS
0438 636 MAC NOVECT MTH MTH$GABS
0438 637 MAC NOVECT MTH MTH$HABS
0438 638 MAC NOVECT MTH MTH$IABS
0438 639 MAC NOVECT MTH MTH$JABS
0438 640 :
0438 641 ; MODULE:MTH$AINT
0438 642 MAC NOVECT MTH MTH$AINT
0438 643 :
0438 644 ; MODULE:MTH$AMOD
0438 645 MAC NOVECT MTH MTH$AMOD
0438 646 :
0438 647 ; MODULE:MTH$ANINT
0438 648 MAC NOVECT MTH MTH$ANINT
0438 649 :
0438 650 ; MODULE:MTH$BITOPS
0438 651 MAC NOVECT MTH MTH$IIAND
0438 652 MAC NOVECT MTH MTH$IIOR
0438 653 MAC NOVECT MTH MTH$IIOR
0438 654 MAC NOVECT MTH MTH$IISHT
0438 655 :
0438 656 MAC NOVECT MTH MTH$INOT
0438 657 MAC NOVECT MTH MTH$JAND
0438 658 MAC NOVECT MTH MTH$JIEOR
0438 659 MAC NOVECT MTH MTH$JIOR
0438 660 :
0438 661 MAC NOVECT MTH MTH$JISHT
0438 662 MAC NOVECT MTH MTH$JNOT
0438 663 :
0438 664 ; MODULE:MTH$CABS
0438 665 MAC CALL MTH MTH$CABS
0440 666 :
0440 667 ; MODULE:MTH$CEXP
0440 668 MAC CALL MTH MTH$CEXP
0448 669 :
0448 670 ; MODULE:MTH$CLOG
0448 671 MAC CALL MTH MTH$CLOG
0450 672 :
0450 673 ; MODULE:MTH$CONJG
0450 674 MAC NOVECT MTH MTH$CONJG
0450 675 :
0450 676 ; MODULE:MTH$CONVER
0450 677 MAC NOVECT MTH MTH$AIMAG
0450 678 MAC NOVECT MTH MTH$DIMAG
0450 679 MAC NOVECT MTH MTH$GIMAG
0450 680 MAC NOVECT MTH MTH$CMPLX
0450 681 MAC NOVECT MTH MTH$DCMPLX
0450 682 MAC NOVECT MTH MTH$GCMPLX
0450 683 MAC NOVECT MTH MTH$DBLE
0450 684 MAC NOVECT MTH MTH$GDBLE
0450 685 MAC NOVECT MTH MTH$DFLOTI
```

0450	686	MAC	NOVECT	MTH	MTH\$DFLOTJ
0450	687	MAC	NOVECT	MTH	MTH\$FLOATI
0450	688	MAC	NOVECT	MTH	MTH\$FLOATJ
0450	689	MAC	NOVECT	MTH	MTH\$GFLOTI
0450	690	MAC	NOVECT	MTH	MTH\$GFLOTJ
0450	691	MAC	NOVECT	MTH	MTH\$IIDINT
0450	692	MAC	NOVECT	MTH	MTH\$IIGINT
0450	693	MAC	NOVECT	MTH	MTH\$IIHINT
0450	694	MAC	NOVECT	MTH	MTH\$IIFIX
0450	695	MAC	NOVECT	MTH	MTH\$IINT
0450	696	MAC	NOVECT	MTH	MTH\$JIDINT
0450	697	MAC	NOVECT	MTH	MTH\$JIGINT
0450	698	MAC	NOVECT	MTH	MTH\$JIHINT
0450	699	MAC	NOVECT	MTH	MTH\$JIFIX
0450	700	MAC	NOVECT	MTH	MTH\$JINT
0450	701	MAC	NOVECT	MTH	MTH\$REAL
0450	702	MAC	NOVECT	MTH	MTH\$DREAL
0450	703	MAC	NOVECT	MTH	MTH\$GREAL
0450	704	MAC	NOVECT	MTH	MTH\$SNGL
0450	705	MAC	NOVECT	MTH	MTH\$SNGLG
0450	706				
0450	707	; MODULE:MTH\$COSH			
0450	708	MAC	CALL	MTH	MTH\$COSH
0458	709				
0458	710	; MODULE:MTH\$CSINCOS			
0458	711	MAC	CALL	MTH	MTH\$CCOS
0460	712	MAC	CALL	MTH	MTH\$CSIN
0468	713				
0468	714	; MODULE:MTH\$CSQRT			
0468	715	MAC	CALL	MTH	MTH\$CSQRT
0470	716				
0470	717	; MODULE:MTH\$DCOSH			
0470	718	MAC	CALL	MTH	MTH\$DCOSH
0478	719				
0478	720	; MODULE:MTH\$DIM			
0478	721	MAC	NOVECT	MTH	MTH\$DDIM
0478	722	MAC	NOVECT	MTH	MTH\$DIM
0478	723	MAC	NOVECT	MTH	MTH\$IIDIM
0478	724	MAC	NOVECT	MTH	MTH\$JIDIM
0478	725				
0478	726	; MODULE:MTH\$DINT			
0478	727	MAC	NOVECT	MTH	MTH\$DINT

0478	729				
0478	730	:	MODULE:MTH\$DMAX1		
0478	731		MAC	NOVECT	MTH
0478	732				MTH\$DMAX1
0478	733	:	MODULE:MTH\$DMIN1		
0478	734		MAC	NOVECT	MTH
0478	735				MTH\$DMIN1
0478	736	:	MODULE:MTH\$DMOD		
0478	737		MAC	NOVECT	MTH
0478	738				MTH\$DMOD
0478	739	:	MODULE:MTH\$DNINT		
0478	740		MAC	NOVECT	MTH
0478	741				MTH\$DNINT
0478	742	:	MODULE:MTH\$DPROD		
0478	743		MAC	NOVECT	MTH
0478	744				MTH\$DPROD
0478	745	:	MODULE:MTH\$DSIGN		
0478	746		MAC	NOVECT	MTH
0478	747				MTH\$DSIGN
0478	748	:	MODULE:MTH\$DSINH		
0478	749		MAC	CALL	MTH
0478	750				MTH\$DSINH
0480	751	:	MODULE:MTH\$DTAN		
0480	752		MAC	CALL	MTH
0480	753				MTH\$DTAN
0488	754	:	MODULE:MTH\$DTANH		
0488	755		MAC	CALL	MTH
0488	756				MTH\$DTANH
0490	757	:	MODULE:MTH\$IIDNNT		
0490	758		MAC	NOVECT	MTH
0490	759				MTH\$IIDNNT
0490	760	:	MODULE:MTH\$IISIGN		
0490	761		MAC	NOVECT	MTH
0490	762				MTH\$IISIGN
0490	763	:	MODULE:MTH\$IMAXO		
0490	764		MAC	NOVECT	MTH
0490	765		MAC	NOVECT	MTH
0490	766				MTH\$AIMAXO
0490	767	:	MODULE:MTH\$IMINO		
0490	768		MAC	NOVECT	MTH
0490	769		MAC	NOVECT	MTH
0490	770				MTH\$IMINO
0490	771	:	MODULE:MTH\$ININT		
0490	772		MAC	NOVECT	MTH
0490	773				MTH\$ININT
0490	774	:	MODULE:MTH\$JIDNNT		
0490	775		MAC	NOVECT	MTH
0490	776				MTH\$JIDNNT
0490	777	:	MODULE:MTH\$JISIGN		
0490	778		MAC	NOVECT	MTH
0490	779				MTH\$JISIGN
0490	780	:	MODULE:MTH\$JMAXO		
0490	781		MAC	NOVECT	MTH
0490	782		MAC	NOVECT	MTH
0490	783				MTH\$AJMAXO
0490	784	:	MODULE:MTH\$JMINO		
0490	785		MAC	NOVECT	MTH
					MTH\$AJMINO

0490	786	MAC	NOVECT	MTH	MTH\$JMIN0
0490	787				
0490	788	; MODULE:MTH\$JNINT			
0490	789	MAC	NOVECT	MTH	MTH\$JNINT
0490	790				
0490	791	; MODULE:MTH\$MAX1			
0490	792	MAC	NOVECT	MTH	MTH\$AMAX1
0490	793	MAC	NOVECT	MTH	MTH\$IMAX1
0490	794	MAC	NOVECT	MTH	MTH\$JMAX1
0490	795				
0490	796	; MODULE:MTH\$MIN1			
0490	797	MAC	NOVECT	MTH	MTH\$AMIN1
0490	798	MAC	NOVECT	MTH	MTH\$IMIN1
0490	799	MAC	NOVECT	MTH	MTH\$JMIN1
0490	800				
0490	801	; MODULE:MTH\$MOD			
0490	802	MAC	NOVECT	MTH	MTH\$IMOD
0490	803	MAC	NOVECT	MTH	MTH\$JMOD
0490	804				
0490	805	; MODULE:MTH\$MSGDEF			
0490	806	MAC	SYM	MTH	MTH\$_FLOOVEMAT
0490	807	MAC	SYM	MTH	MTH\$_FLOUNDMAT
0490	808	MAC	SYM	MTH	MTH\$_INVARGMAT
0490	809	MAC	SYM	MTH	MTH\$_LOGZERNEG
0490	810				
0490	811	MAC	SYM	MTH	MTH\$_SIGLOSMAT
0490	812				
0490	813	MAC	SYM	MTH	MTH\$_SQUROONEG
0490	814	MAC	SYM	MTH	MTH\$_UNDEXP
0490	815	MAC	SYM	MTH	MTH\$_WRONUMARG
0490	816				
0490	817	; MODULE:MTH\$RANDOM			
0490	818	MAC	CALL	MTH	MTH\$RANDOM
0498	819				
0498	820	; MODULE:MTH\$SIGN			
0498	821	MAC	NOVECT	MTH	MTH\$SIGN
0498	822				
0498	823	; MODULE:MTH\$SINH			
0498	824	MAC	CALL	MTH	MTH\$SINH
04A0	825				
04A0	826	; MODULE:MTH\$TAN			
04A0	827	MAC	CALL	MTH	MTH\$TAN
04A8	828				
04A8	829	; MODULE:MTH\$TANH			
04A8	830	MAC	CALL	MTH	MTH\$TANH
04B0	831				

```
04B0 833 :+
04B0 834 : General library entry points LIB$
04B0 835 :-
04B0 836
04B0 837
04B0 838 : MODULE:LIB$AST_IN_PROG
04B0 839 MAC CALL LIB LIB$AST_IN_PROG
04B8 840
04B8 841 : MODULE:LIB$CHAR
04B8 842 MAC NOVECT LIB LIB$CHAR
04B8 843
04B8 844 : MODULE:LIB$CRC
04B8 845 MAC CALL LIB LIB$CRC
04C0 846
04C0 847 : MODULE:LIB$CRC_TABLE
04C0 848 MAC CALL LIB LIB$CRC_TABLE
04C8 849
04C8 850 : MODULE:LIB$CVTDF
04C8 851 MAC NOVECT LIB LIB$CVTDF
04C8 852
04C8 853
04C8 854 : MODULE:LIB$DEC_OVER
04C8 855 MAC CALL LIB LIB$DEC_OVER
04D0 856
04D0 857 : MODULE:LIB$ESTABLISH
04D0 858 MAC CALL LIB LIB$ESTABLISH
04D8 859
04D8 860 : MODULE:LIB$EXTV
04D8 861 MAC CALL LIB LIB$EXTV
04E0 862
04E0 863 : MODULE:LIB$EXTZV
04E0 864 MAC CALL LIB LIB$EXTZV
04E8 865
04E8 866 : MODULE:LIB$FFC
04E8 867 MAC CALL LIB LIB$FFC
04F0 868
04F0 869 : MODULE:LIB$FFS
04F0 870 MAC CALL LIB LIB$FFS
04F8 871
04F8 872 : MODULE:LIB$FIXUP_FLT
04F8 873 MAC CALL LIB LIB$FIXUP_FLT
0500 874
0500 875 : MODULE:LIB$FLT_UNDER
0500 876 MAC CALL LIB LIB$FLT_UNDER
0508 877
0508 878 : MODULE:LIB$GET_INPUT
0508 879 MAC CALL LIB LIB$GET_INPUT
0510 880 MAC CALL LIB LIB$GET_COMMAND
0518 881
0518 882 : MODULE:LIB$ICHR
0518 883 MAC NOVECT LIB LIB$ICHR
0518 884
0518 885 : MODULE:LIB$INDEX
0518 886 MAC CALL LIB LIB$INDEX
0520 887
0520 888 : MODULE:LIB$INITIALIZE
0520 889 MAC NOVECT LIB LIB$INITIALIZE
```

0520	890				
0520	891	; MODULE:LIB\$INSV			
0520	892	MAC	CALL	LIB	LIB\$INSV
0528	893				
0528	894	; MODULE:LIB\$INT_OVER			
0528	895	MAC	CALL	LIB	LIB\$INT_OVER
0530	896				
0530	897	; MODULE:LIB\$LEN			
0530	898	MAC	NOVECT	LIB	LIB\$LEN
0530	899				
0530	900	; MODULE:LIB\$LOCC			
0530	901	MAC	CALL	LIB	LIB\$LOCC
0538	902				
0538	903	; MODULE:LIB\$LOOKUP_KEY			
0538	904	MAC	NOVECT	LIB	LIB\$LOOKUP_KEY
0538	905				
0538	906	; MODULE:LIB\$MATCHC			
0538	907	MAC	CALL	LIB	LIB\$MATCHC
0540	908				
0540	909	; MODULE:LIB\$MATCH_COND			
0540	910	MAC	CALL	LIB	LIB\$MATCH_COND
0548	911				
0548	912	; MODULE:LIB\$MOVTC			
0548	913	MAC	CALL	LIB	LIB\$MOVTC
0550	914				
0550	915	; MODULE:LIB\$MOVTUC			
0550	916	MAC	CALL	LIB	LIB\$MOVTUC
0558	917				
0558	918	; MODULE:LIB\$MSGDEF			
0558	919	MAC	SYM	LIB	LIB\$_AMBKEY
0558	920	MAC	SYM	LIB	LIB\$_ATTCONSTO
0558	921	MAC	SYM	LIB	LIB\$_BADBLOADR
0558	922	MAC	SYM	LIB	LIB\$_BADBLOSIZ
0558	923	MAC	SYM	LIB	LIB\$_BADSTA
0558	924	MAC	SYM	LIB	LIB\$_EF_ALRFRE
0558	925	MAC	SYM	LIB	LIB\$_EF_ALRRES
0558	926	MAC	SYM	LIB	LIB\$_EF_RESSYS
0558	927	MAC	SYM	LIB	LIB\$_FATERRLIB
0558	928	MAC	SYM	LIB	LIB\$_INPSTRTRU
0558	929	MAC	SYM	LIB	LIB\$_INSEF
0558	930	MAC	SYM	LIB	LIB\$_INSVIRMEM
0558	931				
0558	932	MAC	SYM	LIB	LIB\$_INTLOGERR
0558	933	MAC	SYM	LIB	LIB\$_INVARG
0558	934	MAC	SYM	LIB	LIB\$_INVSTRDES
0558	935	MAC	SYM	LIB	LIB\$_NORMAL
0558	936				
0558	937	MAC	SYM	LIB	LIB\$_NOTFOU
0558	938	MAC	SYM	LIB	LIB\$_PUSSTAOVE
0558	939	MAC	SYM	LIB	LIB\$_SIGNO_ARG
0558	940	MAC	SYM	LIB	LIB\$_STRIS_INT
0558	941	MAC	SYM	LIB	LIB\$_STRTRO
0558	942				
0558	943	MAC	SYM	LIB	LIB\$_UNRKEY
0558	944	MAC	SYM	LIB	LIB\$_USEFLORES
0558	945	MAC	SYM	LIB	LIB\$_WRONUMARG
0558	946				

```
0558 947 ; MODULE:LIB$PUT_OUTPUT
0558 948     MAC      CALL    LIB    LIB$PUT_OUTPUT
0560 949
0560 950 ; MODULE:LIB$REVERT
0560 951     MAC      CALL    LIB    LIB$REVERT
0568 952
0568 953 ; MODULE:LIB$SCANC
0568 954     MAC      CALL    LIB    LIB$SCANC
0570 955
0570 956 ; MODULE:LIB$SCOPY
0570 957     MAC      CALL    LIB    LIB$SCOPY_DXDX
0578 958     MAC      JSB     LIB    LIB$SCOPY_DXDX6
0580 959     MAC      CALL    LIB    LIB$SCOPY_R_DX
0588 960     MAC      JSB     LIB    LIB$SCOPY_R_DX6
0590 961     MAC      CALL    LIB    LIB$SGET1_DD
0598 962     MAC      JSB     LIB    LIB$SGET1_DD_R6
05A0 963     MAC      CALL    LIB    LIB$SFREET_DD
05A8 964     MAC      JSB     LIB    LIB$SFREET1_DD6
05B0 965     MAC      CALL    LIB    LIB$SFREEN_DD
05B8 966     MAC      JSB     LIB    LIB$SFREEN_DD6
05C0 967
05C0 968 ; MODULE:LIB$STAT_VM
05C0 969     MAC      CALL    LIB    LIB$STAT_VM
05C8 970
05C8 971 ; MODULE:LIB$SIGNAL
05C8 972     MAC      CALL    LIB    LIB$SIGNAL
05D0 973     MAC      CALL    LIB    LIB$STOP
05D8 974
05D8 975 ; MODULE:LIB$SIG_TO_RET
05D8 976
05D8 977     MAC      CALL    LIB    LIB$SIG_TO_RET
05E0 978
05E0 979 ; MODULE:LIB$SKPC
05E0 980     MAC      CALL    LIB    LIB$SKPC
05E8 981
05E8 982 ; MODULE:LIB$SPANC
05E8 983     MAC      CALL    LIB    LIB$SPANC
05F0 984
05F0 985 ; MODULE:LIB$SYS_ASCTIM
05F0 986     MAC      NOVECT LIB    LIB$SYS_ASCTIM
05F0 987
05F0 988 ; MODULE:LIB$SYS_FAO
05F0 989     MAC      NOVECT LIB    LIB$SYS_FAO
05F0 990
05F0 991 ; MODULE:LIB$SYS_FAOL
05F0 992     MAC      NOVECT LIB    LIB$SYS_FAOL
05F0 993
05F0 994 ; MODULE:LIB$SYS_GETMSG
05F0 995     MAC      NOVECT LIB    LIB$SYS_GETMSG
05F0 996
05F0 997 ; MODULE:LIB$SYS_TRNLOG
05F0 998     MAC      NOVECT LIB    LIB$SYS_TRNLOG
05F0 999
05F0 1000 ; MODULE:LIB$VM
05F0 1001     MAC      CALL    LIB    LIB$FREE_VM
05F8 1002     MAC      CALL    LIB    LIB$GET_VM
0600 1003
```

```
0600 1004 ; MODULE:LIB$STAT VM
0600 1005      MAC      CALL      LIB      LIB$SHOW_VM
0608 1006
0608 1007 ; MODULE:LIB$CURRENCY
0608 1008      MAC      NOVECT      LIB      LIB$CURRENCY
0608 1009
0608 1010 ; MODULE:LIB$DIGIT_SEP
0608 1011      MAC      NOVECT      LIB      LIB$DIGIT_SEP
0608 1012
0608 1013 ; MODULE:LIB$RADIX_POINT
0608 1014      MAC      NOVECT      LIB      LIB$RADIX_POINT
0608 1015
0608 1016 ; MODULE:LIB$RUN_PROGRAM
0608 1017      MAC      NOVECT      LIB      LIB$RUN_PROGRAM
0608 1018
0608 1019 ; MODULE:LIB$DO_COMMAND
0608 1020      MAC      NOVECT      LIB      LIB$DO_COMMAND
0608 1021
0608 1022 ; MODULE:LIB$COMMON
0608 1023      MAC      NOVECT      LIB      LIB$GET_COMMON
0608 1024      MAC      NOVECT      LIB      LIB$PUT_COMMON
0608 1025
0608 1026 ; MODULE:LIB$TRA_ASC_EBC
0608 1027      MAC      NOVECT      LIB      LIB$TRA_ASC_EBC
0608 1028
0608 1029 ; MODULE:LIB$TRA_EBC_ASC
0608 1030      MAC      NOVECT      LIB      LIB$TRA_EBC_ASC
0608 1031
0608 1032 ; MODULE:LIB$INSQHI
0608 1033      MAC      NOVECT      LIB      LIB$INSQHI
0608 1034
0608 1035 ; MODULE:LIB$INSQTI
0608 1036      MAC      NOVECT      LIB      LIB$INSQTI
0608 1037
0608 1038 ; MODULE:LIB$REMQHI
0608 1039      MAC      NOVECT      LIB      LIB$REMQHI
0608 1040
0608 1041 ; MODULE:LIB$REMQTI
0608 1042      MAC      NOVECT      LIB      LIB$REMQTI
0608 1043
```

```
0608 1045 :+
0608 1046 : Internal entry points which need vectors because the non-shared
0608 1047 : library must call these procedures in shared library, rather
0608 1048 : than getting a copy of the procedure from the object library.
0608 1049 : Note: the instances of $$ entry vectors is to be minimized.
0608 1050 : The only cases where it hurts to have two copies of a procedure
0608 1051 : is when the procedure has statically allocated (OWN) data
0608 1052 : which is used as a process-wide resource.
0608 1053 : Note: in order to prevent linker data truncation errors, all modules
0608 1054 : which are shared and are also linked in as private copies when
0608 1055 : $$entry points are referenced by non-shared routines called by the user
0608 1056 : must declare external references to code as general (not word displacement)
0608 1057 : even if the reference is to the same PSECT!!!!
0608 1058 : Modules which have this dual life are: FOR$$ERROR, FOR$$VM, FOR$$SIGNAL.
0608 1059 : *****
0608 1060 : MAINTENANCE NOTE: The following $$ entry vectors can not have their
0608 1061 : specs changed, even though $$, since that would cause user programs
0608 1062 : with compatibility (unshared) routines to have to re-link in order to
0608 1063 : work correctly. Worse we would not want to increase the major ID in
0608 1064 : order for the image activator to catch the incompatibility, since
0608 1065 : that would cause all users to have to relink.
0608 1066 :-
0608 1067
0608 1068 : MODULE:FOR$$CB
0608 1069 :     MAC      JSB      FOR      FOR$$CB_PUSH
0610 1070 :     MAC      JSB      FOR      FOR$$CB_POP
0618 1071 :     MAC      JSB      FOR      FOR$$CB_RET
0620 1072 :     MAC      JSB      FOR      FOR$$CB_GET      ; Added for non-shared code
0628 1073 :                                           ; to load CCB from OTSS$A_CUR_LUB
0628 1074
0628 1075
0628 1076 : MODULE:FOR$ERRSNS
0628 1077 :     MAC      CALL     FOR      FOR$ERRSNS_SAV ; See also above FOR$ERRSNS, FOR$ERRSNS_W
0630 1078
```

```
0630 1080 :+
0630 1081 : Here starts all new entry points defined after VMS 1.00.
0630 1082 : Unless a FUTURE above can be replaced, all new transfer
0630 1083 : points MUST be appended to the end of this list!
0630 1084 :-
0630 1085
0630 1086 MAC CALL FOR FOR$IO_DC_V ; by ref above
0638 1087 MAC CALL FOR FOR$IO_GC_V
0640 1088
0640 1089 : MODULE FOR$CVTRT
0640 1090 MAC CALL FOR FOR$CVT_G_TD
0648 1091 MAC CALL FOR FOR$CVT_G_TE
0650 1092 MAC CALL FOR FOR$CVT_G_TF
0658 1093 MAC CALL FOR FOR$CVT_G_TG
0660 1094
0660 1095 : MODULE OT$SCVTTR
0660 1096 MAC CALL OTS OT$SCVT_T_G
0668 1097
0668 1098 : MODULE FOR$CVTRT
0668 1099 MAC CALL FOR FOR$CVT_H_TD
0670 1100 MAC CALL FOR FOR$CVT_H_TE
0678 1101 MAC CALL FOR FOR$CVT_H_TF
0680 1102 MAC CALL FOR FOR$CVT_H_TG
0688 1103
0688 1104 : MODULE OT$SCVTTR
0688 1105 MAC CALL OTS OT$SCVT_T_H
0690 1106
0690 1107 : MODULE OT$SCVTLT - Old entry points under FOR$
0690 1108 MAC CALL OTS OT$SCVT_L_TI
0698 1109 MAC CALL OTS OT$SCVT_L_TO
06A0 1110 MAC CALL OTS OT$SCVT_L_TZ
06A8 1111 MAC CALL OTS OT$SCVT_L_TL
0680 1112
0680 1113 : MODULE FOR$ENTRY continued from above
0680 1114 MAC CALL FOR FOR$REWRITE_SF FOR$$IO_BEG
0688 1115 MAC CALL FOR FOR$REWRITE_SO FOR$$IO_BEG
06C0 1116 MAC CALL FOR FOR$REWRITE_SU FOR$$IO_BEG
06C8 1117 MAC CALL FOR FOR$READ_IF FOR$$IO_BEG
06D0 1118 MAC CALL FOR FOR$READ_IO FOR$$IO_BEG
06D8 1119 MAC CALL FOR FOR$WRITE_IF FOR$$IO_BEG
06E0 1120 MAC CALL FOR FOR$WRITE_IO FOR$$IO_BEG
06E8 1121
06E8 1122 : MODULE FOR$DELETE
06E8 1123 MAC CALL FOR FOR$DELETE
06F0 1124 MAC CALL FOR FOR$DELETE_D
06F8 1125
06F8 1126 : MODULE FOR$INQUIRE
06F8 1127 MAC CALL FOR FOR$INQUIRE
0700 1128
0700 1129 : MODULE FOR$UNLOCK
0700 1130 MAC CALL FOR FOR$UNLOCK
0708 1131
0708 1132 : MODULE FOR$ENTRY continued
0708 1133 MAC CALL FOR FOR$READ_KU FOR$$IO_BEG
0710 1134
0710 1135 : MODULE FOR$LEX
0710 1136 MAC NOVEC FOR FOR$LGE
```

0710	1137	MAC	NOVECT	FOR	FOR\$LGT
0710	1138	MAC	NOVECT	FOR	FOR\$LLE
0710	1139	MAC	NOVECT	FOR	FOR\$LLT
0710	1140				
0710	1141	; MODULE LIB\$ADDX			
0710	1142	MAC	NOVECT	LIB	LIB\$ADDX
0710	1143	MAC	NOVECT	LIB	LIB\$SUBX
0710	1144				
0710	1145	; MODULE LIB\$ASN_WTH_MBX			
0710	1146	MAC	NOVECT	LIB	LIB\$ASN_WTH_MBX
0710	1147				
0710	1148	; MODULE LIB\$DAY			
0710	1149	MAC	NOVECT	LIB	LIB\$DAY
0710	1150				
0710	1151	; MODULE LIB\$EMODF			
0710	1152	MAC	NOVECT	LIB	LIB\$EMODF
0710	1153				
0710	1154	; MODULE LIB\$EMODD			
0710	1155	MAC	NOVECT	LIB	LIB\$EMODD
0710	1156				
0710	1157	; MODULE LIB\$EMODG			
0710	1158	MAC	NOVECT	LIB	LIB\$EMODG
0710	1159				
0710	1160	; MODULE LIB\$EMODH			
0710	1161	MAC	NOVECT	LIB	LIB\$EMODH
0710	1162				
0710	1163	; MODULE LIB\$EMULATE			
0710	1164	MAC	NOVECT	LIB	LIB\$EMULATE
0710	1165				
0710	1166	; MODULE LIB\$ESTEMU			
0710	1167	MAC	NOVECT	LIB	LIB\$ESTEMU
0710	1168				
0710	1169	; MODULE LIB\$GET_FOREIGN			
0710	1170	MAC	NOVECT	LIB	LIB\$GET_FOREIGN
0710	1171				
0710	1172	; MODULE LIB\$POLYF			
0710	1173	MAC	NOVECT	LIB	LIB\$POLYF
0710	1174				
0710	1175	; MODULE LIB\$POLYD			
0710	1176	MAC	NOVECT	LIB	LIB\$POLYD
0710	1177				
0710	1178	; MODULE LIB\$POLYG			
0710	1179	MAC	NOVECT	LIB	LIB\$POLYG
0710	1180				
0710	1181	; MODULE LIB\$POLYH			
0710	1182	MAC	NOVECT	LIB	LIB\$POLYH
0710	1183				
0710	1184	; MODULE LIB\$SIM_TRAP			
0710	1185	MAC	NOVECT	LIB	LIB\$SIM_TRAP
0710	1186				
0710	1187	; MODULE LIB\$TIMER			
0710	1188	MAC	NOVECT	LIB	LIB\$INIT_TIMER
0710	1189	MAC	NOVECT	LIB	LIB\$SHOW_TIMER
0710	1190	MAC	NOVECT	LIB	LIB\$STAT_TIMER
0710	1191	MAC	NOVECT	LIB	LIB\$FREE_TIMER
0710	1192				
0710	1193	; MODULE MTH\$AINT			

```
0710 1194      MAC      NOVECT  MTH      MTH$SAINT_R2
0710 1195
0710 1196 ; MODULE MTH$SCVTDG
0710 1197      MAC      NOVECT  MTH      MTH$SCVT_D_G
0710 1198      MAC      NOVECT  MTH      MTH$SCVT_G_D
0710 1199
0710 1200 ; MODULE MTH$DFLOOR
0710 1201      MAC      NOVECT  MTH      MTH$DFLOOR
0710 1202      MAC      NOVECT  MTH      MTH$DFLOOR_R3
0710 1203
0710 1204 ; MODULE MTH$DIM
0710 1205      MAC      NOVECT  MTH      MTH$GDIM
0710 1206      MAC      NOVECT  MTH      MTH$HDIM
0710 1207
0710 1208 ; MODULE MTH$DINT
0710 1209      MAC      NOVECT  MTH      MTH$DINT_R4
0710 1210
0710 1211 ; MODULE MTH$DTAN
0710 1212      MAC      JSB      MTH      MTH$DTAN_R7
0718 1213
0718 1214 ; MODULE MTH$FLOOR
0718 1215      MAC      NOVECT  MTH      MTH$FLOOR
0718 1216      MAC      NOVECT  MTH      MTH$FLOOR_R1
0718 1217
0718 1218 ; MODULE MTH$GACOS
0718 1219      MAC      NOVECT  MTH      MTH$GACOS
0718 1220      MAC      NOVECT  MTH      MTH$GACOS_R7
0718 1221
0718 1222 ; MODULE MTH$GASIN
0718 1223      MAC      NOVECT  MTH      MTH$GASIN
0718 1224      MAC      NOVECT  MTH      MTH$GASIN_R7
0718 1225
0718 1226 ; MODULE MTH$GATANH
0718 1227      MAC      NOVECT  MTH      MTH$GATANH
0718 1228
0718 1229 ; MODULE MTH$GCOSH
0718 1230      MAC      NOVECT  MTH      MTH$GCOSH
0718 1231
0718 1232 ; MODULE MTH$GEXP
0718 1233      MAC      NOVECT  MTH      MTH$GEXP
0718 1234      MAC      NOVECT  MTH      MTH$GEXP_R6
0718 1235
0718 1236 ; MODULE MTH$GFLOOR
0718 1237      MAC      NOVECT  MTH      MTH$GFLOOR
0718 1238      MAC      NOVECT  MTH      MTH$GFLOOR_R3
0718 1239
0718 1240 ; MODULE MTH$GINT
0718 1241      MAC      NOVECT  MTH      MTH$GINT
0718 1242      MAC      NOVEC   MTH      MTH$GINT_R4
0718 1243
0718 1244 ; MODULE MTH$GMAX1
0718 1245      MAC      NOVECT  MTH      MTH$GMAX1
0718 1246
0718 1247 ; MODULE MTH$GMIN1
0718 1248      MAC      NOVECT  MTH      MTH$GMIN1
0718 1249
0718 1250 ; MODULE MTH$GMOD
```

0718	1251	MAC	NOVECT	MTH	MTH\$GMOD
0718	1252				
0718	1253	; MODULE MTH\$GNINT			
0718	1254	MAC	NOVECT	MTH	MTH\$GNINT
0718	1255				
0718	1256	; MODULE MTH\$GPROD			
0718	1257	MAC	NOVECT	MTH	MTH\$GPROD
0718	1258				
0718	1259	; MODULE MTH\$GSIGN			
0718	1260	MAC	NOVECT	MTH	MTH\$GSIGN
0718	1261				
0718	1262	; MODULE MTH\$GSINCOS			
0718	1263	MAC	NOVECT	MTH	MTH\$GSIN
0718	1264	MAC	NOVECT	MTH	MTH\$GCOS
0718	1265	MAC	NOVECT	MTH	MTH\$GSIN_R7
0718	1266	MAC	NOVECT	MTH	MTH\$GCOS_R7
0718	1267				
0718	1268	; MODULE MTH\$GSINH			
0718	1269	MAC	NOVECT	MTH	MTH\$GSINH
0718	1270				
0718	1271	; MODULE MTH\$GSQRT			
0718	1272	MAC	NOVECT	MTH	MTH\$GSQRT
0718	1273	MAC	NOVECT	MTH	MTH\$GSQRT_R5
0718	1274				
0718	1275	; MODULE MTH\$GTAN			
0718	1276	MAC	NOVECT	MTH	MTH\$GTAN
0718	1277	MAC	NOVECT	MTH	MTH\$GTAN_R7
0718	1278				
0718	1279	; MODULE MTH\$GTANH			
0718	1280	MAC	NOVECT	MTH	MTH\$GTANH
0718	1281				
0718	1282	; MODULE MTH\$HACOS			
0718	1283	MAC	NOVECT	MTH	MTH\$HACOS
0718	1284	MAC	NOVECT	MTH	MTH\$HACOS_R8
0718	1285				
0718	1286	; MODULE MTH\$HASIN			
0718	1287	MAC	NOVECT	MTH	MTH\$HASIN
0718	1288	MAC	NOVECT	MTH	MTH\$HASIN_R8
0718	1289				
0718	1290	; MODULE MTH\$HCOSH			
0718	1291	MAC	NOVECT	MTH	MTH\$HCOSH
0718	1292				
0718	1293	; MODULE MTH\$HEXP			
0718	1294	MAC	NOVECT	MTH	MTH\$HEXP
0718	1295	MAC	NOVECT	MTH	MTH\$HEXP_R6
0718	1296				
0718	1297	; MODULE MTH\$HFLOOR			
0718	1298	MAC	NOVECT	MTH	MTH\$HFLOOR
0718	1299	MAC	NOVECT	MTH	MTH\$HFLOOR_R7
0718	1300				
0718	1301	; MODULE MTH\$HINT			
0718	1302	MAC	NOVECT	MTH	MTH\$HINT
0718	1303	MAC	NOVECT	MTH	MTH\$HINT_R8
0718	1304	; MODULE MTH\$HMAX1			
0718	1305	MAC	NOVECT	MTH	MTH\$HMAX1
0718	1306				
0718	1307	; MODULE MTH\$HMIN1			

```
0718 1308          MAC      NOVECT  MTH      MTH$HMIN1
0718 1309
0718 1310 ; MODULE MTH$HMOD
0718 1311          MAC      NOVECT  MTH      MTH$HMOD
0718 1312
0718 1313 ; MODULE MTH$HNINT
0718 1314          MAC      NOVECT  MTH      MTH$HNINT
0718 1315
0718 1316 ; MODULE MTH$HSIGN
0718 1317          MAC      NOVECT  MTH      MTH$HSIGN
0718 1318
0718 1319 ; MODULE MTH$HSINCOS
0718 1320          MAC      NOVECT  MTH      MTH$HSIN
0718 1321          MAC      NOVECT  MTH      MTH$HSIN_R5
0718 1322          MAC      NOVECT  MTH      MTH$HCOS
0718 1323          MAC      NOVECT  MTH      MTH$HCOS_R5
0718 1324
0718 1325 ; MODULE MTH$HSINH
0718 1326          MAC      NOVECT  MTH      MTH$HSINH
0718 1327
0718 1328 ; MODULE MTH$HSQRT
0718 1329          MAC      NOVECT  MTH      MTH$HSQRT
0718 1330          MAC      NOVECT  MTH      MTH$HSQRT_R8
0718 1331
0718 1332 ; MODULE MTH$HTAN
0718 1333          MAC      NOVECT  MTH      MTH$HTAN
0718 1334          MAC      NOVECT  MTH      MTH$HTAN_R5
0718 1335
0718 1336 ; MODULE MTH$HTANH
0718 1337          MAC      NOVECT  MTH      MTH$HTANH
0718 1338
0718 1339 ; MODULE MTH$IIGNNT
0718 1340          MAC      NOVECT  MTH      MTH$IIGNNT
0718 1341
0718 1342 ; MODULE MTH$IIHNNT
0718 1343          MAC      NOVECT  MTH      MTH$IIHNNT
0718 1344
0718 1345 ; MODULE MTH$JIGNNT
0718 1346          MAC      NOVECT  MTH      MTH$JIGNNT
0718 1347
0718 1348 ; MODULE MTH$JIHNNT
0718 1349          MAC      NOVECT  MTH      MTH$JIHNNT
0718 1350
0718 1351 ; MODULE MTH$STAN
0718 1352          MAC      JSB      MTH      MTH$STAN_R4
0720 1353
0720 1354 ; MODULE MTH$SGN
0720 1355          MAC      NOVECT  MTH      MTH$SGN
0720 1356          MAC      NOVECT  MTH      MTH$SGN_R1
0720 1357
0720 1358 ; MODULE OTS$POWGG
0720 1359          MAC      NOVECT  OTS      OTS$POWGG
0720 1360
0720 1361 ; MODULE OTS$POWGJ
0720 1362          MAC      NOVECT  OTS      OTS$POWGJ
0720 1363
0720 1364 ; MODULE OTS$POWHJ
```

0720	1365	MAC	NOVECT	OTS	OTSS\$POWHJ_R3
0720	1366				
0720	1367	; MODULE OTSS\$DIVCD			
0720	1368	MAC	NOVECT	OTS	OTSS\$DIVCD_R3
0720	1369				
0720	1370	; MODULE OTSS\$DIVCG			
0720	1371	MAC	NOVECT	OTS	OTSS\$DIVCG_R3
0720	1372				
0720	1373	; MODULE OTSS\$MULCD			
0720	1374	MAC	NOVECT	OTS	OTSS\$MULCD_R3
0720	1375				
0720	1376	; MODULE OTSS\$MULCG			
0720	1377	MAC	NOVECT	OTS	OTSS\$MULCG_R3
0720	1378				
0720	1379	; MODULE MTH\$CDABS			
0720	1380	MAC	NOVECT	MTH	MTH\$CDABS
0720	1381				
0720	1382	; MODULE MTH\$DCONJG			
0720	1383	MAC	NOVECT	MTH	MTH\$DCONJG
0720	1384				
0720	1385	; MODULE MTH\$CDEXP			
0720	1386	MAC	NOVECT	MTH	MTH\$CDEXP
0720	1387				
0720	1388	; MODULE MTH\$CDLOG			
0720	1389	MAC	NOVECT	MTH	MTH\$CDLOG
0720	1390				
0720	1391	; MODULE MTH\$CDSINCOS			
0720	1392	MAC	NOVECT	MTH	MTH\$CDCOS
0720	1393	MAC	NOVECT	MTH	MTH\$CDSIN
0720	1394				
0720	1395	; MODULE MTH\$CDSQRT			
0720	1396	MAC	NOVECT	MTH	MTH\$CDSQRT
0720	1397				
0720	1398	; MODULE MTH\$CGABS			
0720	1399	MAC	NOVECT	MTH	MTH\$CGABS
0720	1400				
0720	1401	; MODULE MTH\$GCONJG			
0720	1402	MAC	NOVECT	MTH	MTH\$GCONJG
0720	1403				
0720	1404	; MODULE MTH\$CGEXP			
0720	1405	MAC	NOVECT	MTH	MTH\$CGEXP
0720	1406				
0720	1407	; MODULE MTH\$CGLOG			
0720	1408	MAC	NOVECT	MTH	MTH\$CGLOG
0720	1409				
0720	1410	; MODULE MTH\$CGSINCOS			
0720	1411	MAC	NOVECT	MTH	MTH\$CGCOS
0720	1412	MAC	NOVECT	MTH	MTH\$CGSIN
0720	1413				
0720	1414	; MODULE MTH\$CGSQRT			
0720	1415	MAC	NOVECT	MTH	MTH\$CGSQRT
0720	1416				
0720	1417	; MODULE OTSS\$POWCC			
0720	1418	MAC	NOVECT	OTS	OTSS\$POWCC
0720	1419				
0720	1420	; MODULE OTSS\$POWCDCD			
0720	1421	MAC	NOVECT	OTS	OTSS\$POWCDCD_R3

```
0720 1422
0720 1423 ; MODULE OTS$POWCDJ
0720 1424 MAC NOVECT OTS OTS$POWCDJ_R3
0720 1425
0720 1426 ; MODULE OTS$POWCGCG
0720 1427 MAC NOVECT OTS OTS$POWCGCG_R3
0720 1428
0720 1429 ; MODULE OTS$POWCGJ
0720 1430 MAC NOVECT OTS OTS$POWCGJ_R3
0720 1431
0720 1432 ; MODULE OTS$POWHH
0720 1433 MAC NOVECT OTS OTS$POWHH_R3
0720 1434
0720 1435 ; MODULE MTH$SQRT
0720 1436 MAC JSB MTH MTH$SQRT_R3
0728 1437
0728 1438 ;+
0728 1439 ; The following routine is called from FOR$$ERROR, therefore
0728 1440 ; it must be vectored.
0728 1441 ; -
0728 1442
0728 1443 ; MODULE FOR$$CB (continued)
0728 1444 MAC CALL FOR FOR$$FP_MATCH
0730 1445
0730 1446 ; MODULE FOR$READ SN
0730 1447 MAC CALL FOR FOR$READ_SN FOR$$IO_BEG
0738 1448
0738 1449 ; MODULE FOR$WRITE SN
0738 1450 MAC CALL FOR FOR$WRITE_SN FOR$$IO_BEG
0740 1451
0740 1452 ; MODULE FOR$IO_ELEM (continued)
0740 1453 MAC CALL FOR FOR$IO_X_SB
0748 1454 MAC CALL FOR FOR$IO_X_NL
0750 1455 MAC CALL FOR FOR$IO_X_SE
0758 1456
0758 1457 ; MODULE OTS$CVTLT (continued)
0758 1458 MAC CALL OTS OTS$CVT_L_TB
0760 1459
0760 1460 ; MODULE OTS$CVTTOL (continued)
0760 1461 MAC CALL OTS OTS$CVT_TB_L
0768 1462
0768 1463 ; MODULE OTS$CVTTF
0768 1464 MAC CALL OTS OTS$CVT_T_F
0770 1465
0770 1466 ; MODULE LIB$ATTACH
0770 1467 MAC CALL LIB LIB$ATTACH
0778 1468
0778 1469 ; MODULE LIB$SPAWN
0778 1470 MAC CALL LIB LIB$SPAWN
0780 1471
0780 1472 ; MODULE LIB$GET_OPCODE
0780 1473 MAC CALL LIB LIB$GET_OPCODE
0788 1474
0788 1475 ; MODULE FOR$RAB
0788 1476 MAC CALL FOR FOR$RAB
0790 1477
0790 1478 ;+
```

```
0790 1479 : The following three entry points are for the 'kernel' floating output
0790 1480 : conversion routines. Although they are 'double-dollar' names, they are
0790 1481 : vectored so that future language-specific shareable images can use them.
0790 1482 :-
0790 1483
0790 1484 : MODULE OTS$$CVTDT
0790 1485 :     MAC      JSB      OTS      OTS$$CVT_D_T_R8
0798 1486
0798 1487 : MODULE OTS$$CVTRT
0798 1488 :     MAC      JSB      OTS      OTS$$CVT_G_T_R8
07A0 1489 :     MAC      JSB      OTS      OTS$$CVT_H_T_R8
07A8 1490
07A8 1491 :+
07A8 1492 : The following entries are present only so that there will be references
07A8 1493 : to these symbols in this module. Without them, the linker complains.
07A8 1494 : Note that these entries are not universal, so they cannot be referenced
07A8 1495 : by those linking to this image.
07A8 1496 :-
07A8 1497
0000' 07A8 1498      .MASK  BAS$$HANDLER
00000000'GF 17 07AA 1499      JMP      G^BAS$$HANDLER
0000' 07B0 1500      .MASK  COB$$HANDLER
00000000'GF 17 07B2 1501      JMP      G^COB$$HANDLER
0000' 07B8 1502      .MASK  FOR$$IO_BEG
00000000'GF 17 07BA 1503      JMP      G^FOR$$IO_BEG
07C0 1504
```

[illegible]

```
0800 1517 :+
0800 1518 : The following entry points are for the string library.
0800 1519 : The JSB entry points are later.
0800 1520 :-
0800 1521 MAC CALL STR STR$CONCAT ; Concatenate strings
0808 1522 MAC CALL STR STR$COPY_DX ; Copy by descriptor
0810 1523 MAC CALL STR STR$COPY_R ; Copy by reference
0818 1524 MAC CALL STR STR$FREE_DX ; Free a string
0820 1525 MAC CALL STR STR$GET1_DX ; Allocate a string
0828 1526 MAC NOVECT STR STR$ADD ; Add two strings
0828 1527 MAC NOVECT STR STR$MUL ; Multiply two strings
0828 1528 MAC NOVECT STR STR$RECIP ; Take 1/ a string
0828 1529 MAC NOVECT STR STR$ROUND ; Arithmetically round a string
0828 1530 MAC CALL STR STR$LEFT ; Take left part of string
0830 1531 MAC NOVECT STR STR$LEFT_R8 ; (JSB entry point)
0830 1532 MAC CALL STR STR$LEN_EXTR ; Extract from a string by l
0838 1533 MAC NOVECT STR STR$LEN_EXTR_R8 ; (JSB entry point)
0838 1534 MAC CALL STR STR$POS_EXTR ; Extract from a string by p
0840 1535 MAC NOVECT STR STR$POS_EXTR_R8 ; (JSB entry point)
0840 1536 MAC CALL STR STR$POSITION ;
0848 1537 MAC NOVECT STR STR$POSITION_R6 ; (JSB entry point)
0848 1538 MAC CALL STR STR$RIGHT ; Take right part of a string
0850 1539 MAC NOVECT STR STR$RIGHT_R8 ; (JSB entry point)
0850 1540 MAC CALL STR STR$DUPL_CHAR ; Make lots of a character
0858 1541 MAC CALL STR STR$TRIM ; Remove trailing blanks
0860 1542 MAC FUTURE STR STR$FUTURE_1 ; Reserved for future expansions
0868 1543 MAC FUTURE STR STR$FUTURE_2
0870 1544 MAC FUTURE STR STR$FUTURE_3
0878 1545 :+
0878 1546 : String condition codes
0878 1547 :-
0878 1548 MAC SYM STR STR$_DIVBY_ZER ; Divide by zero
0878 1549 MAC SYM STR STR$_FATINTERR ; Fatal internal error
0878 1550 MAC SYM STR STR$_ILLSTRCLA ; Illegal string class
0878 1551 MAC SYM STR STR$_ILLSTRPOS
0878 1552 MAC SYM STR STR$_ILLSTRSPE
0878 1553 MAC SYM STR STR$_INSVIRMEM
0878 1554 MAC SYM STR STR$_MATCH
0878 1555 MAC SYM STR STR$_NEGSTRLEN
0878 1556 MAC SYM STR STR$_NOMATCH
0878 1557 MAC SYM STR STR$_NORMAL
0878 1558 MAC SYM STR STR$_STRIS_INT
0878 1559 MAC SYM STR STR$_STRTOOLON
0878 1560 MAC SYM STR STR$_TRU
0878 1561 MAC SYM STR STR$_WRONUMARG
0878 1562
```

```
0878 1564 :+
0878 1565 : The following entry points are generated by the BASIC-PLUS-2
0878 1566 : compiler. The current arrangement of which are vectored is tentative.
0878 1567 :-
0878 1568 :
0878 1569 :
0878 1570 :
0878 1571 :
0878 1572 :
0880 1573 :
0880 1574 :
0888 1575 :
0888 1576 :
0888 1577 :
0888 1578 :
0888 1579 :
0888 1580 :
0888 1581 :
0890 1582 :
0898 1583 :
0898 1584 :
0898 1585 :
0898 1586 :
0898 1587 :
```

ARITHMETIC CODE SUPPORT

MAC	NOVECT	BAS	BASS\$POWII	: Integer(w) ** Integer(w)
MAC	JSB	BAS	BASS\$SCALE_D_R1	: Scale a number
MAC	NOVECT	BAS	BASS\$POWJJ	: Long ** Long
MAC	JSB	BAS	BASS\$DS\$SCALE_D_R1	: Descal a number
MAC	NOVECT	BAS	BASS\$POWRJ	: Float ** Long
MAC	NOVECT	BAS	BASS\$POWRR	: Float ** Float
MAC	NOVECT	BAS	BASS\$POWDJ	: Double ** Long
MAC	NOVECT	BAS	BASS\$POWDD	: Double ** Double
MAC	NOVECT	BAS	BASS\$RND_F_R1	: Return random number
MAC	NOVECT	BAS	BASS\$RANDOMIZE	: Perturb the random seed
MAC	CALL	BAS	BASS\$CMPF_APP	: Approximate float compare
MAC	CALL	BAS	BASS\$CMPD_APP	: Approximate double compar

STRING CODE SUPPORT

MAC	NOVECT	BAS	BASS\$CHANGE_NA_S	: CHANGE A% to A\$
MAC	NOVECT	BAS	BASS\$CHANGE_S_NA	: CHANGE A\$ to A%

0898	1589	:					
0898	1590	:					
0898	1591	:					
0898	1592	:					
08A0	1593	:					
08A8	1594	:					
08B0	1595	:					
08B0	1596	:					
08B0	1597	:					
08B8	1598	:					
08B8	1599	:					
08B8	1600	:					
08B8	1601	:					
08C0	1602	:					
08C8	1603	:					
08D0	1604	:					
08D8	1605	:					
08E0	1606	:					
08E8	1607	:					
08F0	1608	:					
08F8	1609	:					
0900	1610	:					
0900	1611	:					
0900	1612	:					
0908	1613	:					
0910	1614	:					
0918	1615	:					

STRING FUNCTIONS					
MAC	CALL	BAS	BASSRSET	:	String move, right justif
MAC	CALL	BAS	BASSRSET_R	:	By-ref entry point
MAC	CALL	BAS	BASSEDIT	:	String editing
MAC	NOVECT	BAS	BASSFORMAT_F	:	Floating FORMATS
MAC	NOVECT	BAS	BASSFORMAT_D	:	Double FORMATS
MAC	CALL	BAS	BASSINSTR	:	Match substring
MAC	NOVECT	BAS	BASSRAD	:	RADIX 50
MAC	NOVECT	BAS	BASSRAD50	:	RADIX 50
MAC	NOVECT	BAS	BASSSTOP	:	STOP statement (** here
MAC	CALL	BAS	BASSSTR_F	:	Return binary->ASCII numb
MAC	CALL	BAS	BASSSTR_D	:	Ditto for double
MAC	CALL	BAS	BASSSTR_L	:	Ditto for longword
MAC	CALL	BAS	BASSNUM_F	:	Return binary->ASCII numb
MAC	CALL	BAS	BASSNUM_D	:	Ditto for double
MAC	CALL	BAS	BASSNUM_L	:	Ditto for longword
MAC	CALL	BAS	BASSNUMT_F	:	Return binary->ASCII numb
MAC	CALL	BAS	BASSNUM1_D	:	Ditto for double
MAC	CALL	BAS	BASSNUM1_L	:	Ditto for longword
MAC	NOVECT	BAS	BASSTAB	:	Tab over x spaces
MAC	NOVECT	BAS	BASSTIME T	:	24 hour time string
MAC	CALL	BAS	BASSVAL_C	:	Return ASCII->binary stri
MAC	CALL	BAS	BASSVAL_F	:	Ditto for floating
MAC	CALL	BAS	BASSVAL_D	:	Ditto for double
MAC	NOVECT	BAS	BASSXLATE	:	Translate a string

```
0918 1617 :  
0918 1618 :  
0918 1619 :  
0918 1620 :  
0918 1621 :  
0918 1622 :  
0918 1623 :  
0918 1624 :  
0918 1625 :  
0918 1626 :  
0918 1627 :  
0918 1628 :  
0918 1629 :  
0920 1630 :  
0928 1631 :  
0930 1632 :  
0938 1633 :  
0938 1634 :  
0940 1635 :  
0948 1636 :  
0950 1637 :  
0958 1638 :  
0958 1639 :  
0958 1640 :  
0958 1641 :  
0960 1642 :  
0968 1643 :  
0970 1644 :  
0978 1645 :  
0980 1646 :  
0988 1647 :  
0990 1648 :  
0998 1649 :  
0998 1650 :
```

STRING ARITHMETIC

MAC	NOVECT	BAS	BASSCOMP	: String arith compare
MAC	NOVECT	BAS	BASSDIF	: S.A. difference
MAC	NOVECT	BAS	BASSPLACE	: S.A. precision
MAC	NOVECT	BAS	BASSPROD	: S.A. multiplication
MAC	NOVECT	BAS	BASSQUO	: S.A. division
MAC	NOVECT	BAS	BASSSUM	: S.A. addition

PROCEDURE ACTIVATION

MAC	JSB	BAS	BASSINIT_R8	: Main program initializer
MAC	JSB	BAS	BASSINIT_DEF_R8	: DEF function initializer
MAC	JSB	BAS	BASSINIT_DFS_R8	: DEF* function initializer
MAC	CALL	BAS	BASSINIT_GOSUB	: GOSUB initializer
MAC	NOVECT	BAS	BASSINIT_C_GSB	: ON-GOSUB initializer
MAC	JSB	BAS	BASSEND_R8	: Main program ender
MAC	JSB	BAS	BASSEND_DEF_R8	: DEF function end
MAC	JSB	BAS	BASSEND_DFS_R8	: DEF* function end
MAC	JSB	BAS	BASSEND_GSB_R8	: GOSUB end

ERROR HANDLING

MAC	CALL	BAS	BASSON_ERR_Z	: ON ERROR GOTO 0
MAC	CALL	BAS	BASSON_ERR_BK	: ON ERROR GOBACK
MAC	CALL	BAS	BASSRESUME	: RESUME line num
MAC	CALL	BAS	BASSRESUME_Z	: RESUME
MAC	CALL	BAS	BASSERR	: ERR variable
MAC	CALL	BAS	BASSERL	: ERL variable
MAC	CALL	BAS	BASSERN	: ERN\$ variable
MAC	CALL	BAS	BASSERT	: ERT\$ variable

MAC	CALL	BAS	BASSHANDLER	BASS\$HANDLER
-----	------	-----	-------------	---------------

```
09A0 1652 :  
09A0 1653 : Scalar (non matrix) I/O  
09A0 1654 :  
09A0 1655 MAC CALL BAS BASSINPUT : Initialize for INPUT unit  
09A8 1656 MAC CALL BAS BASSINPUT : Ditto LINPUT unit  
09B0 1657 MAC CALL BAS BASSINPUT_LINE : Ditto INPUT LINE unit  
09B8 1658 MAC CALL BAS BASSREAD : Ditto READ  
09C0 1659 MAC CALL BAS BASSPRINT : Ditto PRINT  
09C8 1660 MAC CALL BAS BASSPRINT USING : Ditto PRINT USING  
09D0 1661 MAC CALL BAS BASSIO_END : End of I/O list  
09D8 1662 MAC CALL BAS BASSIN_W_R : INPUT word  
09E0 1663 MAC CALL BAS BASSIN_L_R : INPUT long  
09E8 1664 MAC CALL BAS BASSIN_F_R : INPUT float  
09F0 1665 MAC CALL BAS BASSIN_D_R : INPUT double  
09F8 1666 MAC CALL BAS BASSIN_T_DX : INPUT string  
0A00 1667 MAC CALL BAS BASSOUT_C_V_S : PRINT long(word) ;  
0A08 1668 MAC CALL BAS BASSOUT_L_V_B : PRINT long(word)  
0A10 1669 MAC CALL BAS BASSOUT_L_V_C : PRINT long(word) ,  
0A18 1670 MAC CALL BAS BASSOUT_F_V_S : PRINT float ;  
0A20 1671 MAC CALL BAS BASSOUT_F_V_B : PRINT float  
0A28 1672 MAC CALL BAS BASSOUT_F_V_C : PRINT float ,  
0A30 1673 MAC CALL BAS BASSOUT_D_V_S : PRINT double ;  
0A38 1674 MAC CALL BAS BASSOUT_D_V_B : PRINT double  
0A40 1675 MAC CALL BAS BASSOUT_D_V_C : PRINT double ,  
0A48 1676 MAC CALL BAS BASSOUT_T_DX_S : PRINT string ;  
0A50 1677 MAC CALL BAS BASSOUT_T_DX_B : PRINT string  
0A58 1678 MAC CALL BAS BASSOUT_T_DX_C : PRINT string ,  
0A60 1679 :  
0A60 1680 : Matrix I/O  
0A60 1681 :  
0A60 1682 MAC NOVECT BAS BASSOUT_MAT_S : Output element xmtr  
0A60 1683 MAC NOVECT BAS BASSOUT_MAT_B :  
0A60 1684 MAC NOVECT BAS BASSOUT_MAT_C :  
0A60 1685 MAC NOVECT BAS BASSIN_MAT : Input element xmtr  
0A60 1686 MAC CALL BAS BASSMAT_PRINT : Init for MAT PRINT  
0A68 1687 MAC CALL BAS BASSMAT_INPUT : for MAT INPUT  
0A70 1688 MAC CALL BAS BASSMAT_LINPUT : for MAT LINPUT  
0A78 1689 MAC CALL BAS BASSMAT_READ : for MAT READ  
0A80 1690 MAC NOVECT BAS BASSNUM : NUM var (mat)  
0A80 1691 MAC NOVECT BAS BASSNUM2 : NUM2 var (mat)  
0A80 1692 :  
0A80 1693 :  
0A80 1694 : RMS I/O  
0A80 1695 :  
0A80 1696 MAC CALL BAS BASSOPEN : OPEN (all)  
0A88 1697 MAC CALL BAS BASSCLOSE : CLOSE (all)  
0A90 1698 MAC CALL BAS BASSGET : Sequential GET  
0A98 1699 MAC CALL BAS BASSGET_RECORD : Random GET  
0AA0 1700 MAC CALL BAS BASSGET_KEY : Indexed GET  
0AA8 1701 MAC CALL BAS BASSPUT : Sequential PUT  
0AB0 1702 MAC CALL BAS BASSPUT_RECORD : Random PUT  
0AB8 1703 MAC CALL BAS BASSPUT_COUNT : Sequential PUT w/COUNT  
0AC0 1704 MAC CALL BAS BASSPUT_REC_CNT : Random PUT w/COUNT  
0AC8 1705 MAC CALL BAS BASSFIND : Sequential FIND  
0AD0 1706 MAC CALL BAS BASSFIND_RECORD : Random FIND  
0AD8 1707 MAC CALL BAS BASSFIND_KEY : Indexed FIND  
0AE0 1708 MAC CALL BAS BASSDELETE : DELETE
```

VMSSVECTOR
4-003

- Define entry vectors for VMSRTL H 10

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 37
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (20)

OAE8	1709	MAC	CALL	BAS	BASSUPDATE	:	UPDATE
OAF0	1710	MAC	CALL	BAS	BASSUPDATE COUN	:	UPDATE w/COUNT
OAF8	1711	MAC	CALL	BAS	BASSRESTORE	:	RESTORE
OB00	1712	MAC	CALL	BAS	BASSRESTORE_KEY	:	Indexed RESTORE
OB08	1713	MAC	CALL	BAS	BASSCRATCH	:	SCRATCH
OB10	1714	MAC	CALL	BAS	BASSUNLOCK	:	Release
OB18	1715	MAC	CALL	BAS	BASSFREE	:	FREE

```
OB20 1717 :  
OB20 1718 :  
OB20 1719 :  
OB20 1720 :  
OB20 1721 :  
OB20 1722 :  
OB20 1723 :  
OB20 1724 :  
OB20 1725 :  
OB20 1726 :  
OB20 1727 :  
OB20 1728 :  
OB20 1729 :  
OB20 1730 :  
OB20 1731 :  
OB20 1732 :  
OB20 1733 :  
OB20 1734 :  
OB28 1735 :  
OB28 1736 :  
OB28 1737 :  
OB28 1738 :  
OB28 1739 :  
OB28 1740 :  
OB30 1741 :  
OB38 1742 :  
OB40 1743 :  
OB40 1744 :  
OB40 1745 :  
OB40 1746 :  
OB40 1747 :  
OB40 1748 :  
OB48 1749 :  
OB48 1750 :  
OB48 1751 :  
OB48 1752 :  
OB48 1753 :  
OB48 1754 :  
OB48 1755 :  
OB48 1756 :  
  
RSTS/E COMPATIBILITY  
  
MAC NOVECT BAS BASSCVT-W-S : CVT%$  
MAC NOVECT BAS BASSCVT-S-W : CVT$%  
MAC NOVECT BAS BASSCVT-F-S : CVTF$  
MAC NOVECT BAS BASSCVT-D-S : CVTD$  
MAC NOVECT BAS BASSCVT-S-F : CVTSF  
MAC NOVECT BAS BASSCVT-S-D : CVTSD  
MAC NOVECT BAS BASSFSS : File string scan  
MAC NOVECT BAS BASSFSP : File info  
MAC NOVECT BAS BASSSYS : Sys calls  
MAC NOVECT BAS BASSCHAIN : CHAIN statement  
MAC NOVECT BAS BASSPEEK : Examine RSTS/E memory  
  
MISC  
  
MAC CALL BAS BASSCCPOS : CCPOS func  
MAC NOVECT BAS BASSECHO : Enable echo  
MAC NOVECT BAS BASSONECHR : *****  
MAC NOVECT BAS BASSNOECHO : Disable echo  
MAC NOVECT BAS BASSRCTRL0 : Disable CTRL 0  
MAC NOVECT BAS BASSCTRL0 : Cause control 0  
MAC CALL BAS BASSRESTORE_DAT : RESTORE data  
MAC CALL BAS BASSRECOUNT : RECOUNT var  
MAC CALL BAS BASSSTATUS : STATUS var  
MAC NOVECT BAS BASSMAGTAPE : MAGTAPE func  
MAC NOVECT BAS BAS$TIME F : Floating time values  
MAC NOVECT BAS BASS$SLEEP : SLEEP statement  
MAC NOVECT BAS BASSNAME_AS : NAME AS statement  
MAC NOVECT BAS BASSKILL : KILL erase  
MAC CALL BAS BASSBUFSIZ : BUFSIZ func  
  
MOVE STATEMENT  
  
MAC NOVECT BAS BASS$MOVE_TO : Start a MOVE TO statement  
MAC NOVECT BAS BASS$MOVE_FROM : Start a MOVE FROM statemen  
MAC NOVECT BAS BASS$MOVE_END : End of a MOVE statement  
MAC NOVECT BAS BASS$MOVE_ARRAY : MOVE an array
```

OB48	1758	:					
OB48	1759	:					
OB48	1760	:					
OB48	1761	:	MAC	NOVECT	BAS	BASSDET_F	: Determinate
OB48	1762	:	MAC	NOVECT	BAS	BASSDET_D	: Determinate
OB48	1763	:	MAC	NOVECT	BAS	BASSMAT_NULL	: Null out string matrix
OB48	1764	:	MAC	NOVECT	BAS	BASSMAT_ASSIGN	: Matrix assignments
OB48	1765	:	MAC	NOVECT	BAS	BASSMAT_INIT	: Matrix initialize(0 or 1)
OB48	1766	:	MAC	NOVECT	BAS	BASSMAT_IDN	: Matrix identity
OB48	1767	:	MAC	NOVECT	BAS	BASSMAT_ADD	: Matrix addition
OB48	1768	:	MAC	NOVECT	BAS	BASSMAT_SUB	: Matrix subtraction
OB48	1769	:	MAC	NOVECT	BAS	BASSMAT_MUL	: Matrix multiplication
OB48	1770	:	MAC	NOVECT	BAS	BASSMAT_SCA_MUL	: Matrix scalar multiplicat
OB48	1771	:	MAC	NOVECT	BAS	BASSMAT_TRN	: Matrix transposition
OB48	1772	:	MAC	NOVECT	BAS	BASSMAT_INV	: Matrix inversion
OB48	1773	:	MAC	NOVECT	BAS	BASSMAT_REDIM	: Single redimension
OB48	1774	:					
OB48	1775	:					
OB48	1776	:					
OB48	1777	:	MAC	CALL	BAS	BASSCHR	: Return character for bina
OB50	1778	:					
OB50	1779	:					
OB50	1780	:					
OB50	1781	:	MAC	NOVECT	BAS	BASSFET_FA_W_R8	: Fetch a word from virtual
OB50	1782	:	MAC	NOVECT	BAS	BASSFET_FA_L_R8	
OB50	1783	:	MAC	NOVECT	BAS	BASSFET_FA_F_R8	
OB50	1784	:	MAC	NOVECT	BAS	BASSFET_FA_D_R8	
OB50	1785	:	MAC	NOVECT	BAS	BASSFETCH_BFA	
OB50	1786	:	MAC	NOVECT	BAS	BASSSTO_FA_W_R8	: Store a word in a virtual
OB50	1787	:	MAC	NOVECT	BAS	BASSSTO_FA_L_R8	
OB50	1788	:	MAC	NOVECT	BAS	BASSSTO_FA_F_R8	
OB50	1789	:	MAC	NOVECT	BAS	BASSSTO_FA_D_R8	
OB50	1790	:	MAC	NOVECT	BAS	BASSSTORE_BFA	
OB50	1791	:	MAC	NOVECT	BAS	BASSSTO_FA_RDX	

OB50	1793	:				
OB50	1794	:				
OB50	1795	:				
OB50	1796	:	MAC	NOVECT	BAS	BASSFIELD_SET
OB50	1797	:	MAC	NOVECT	BAS	BASSFIELD_COPY
OB50	1798	:	MAC	NOVECT	BAS	BASSFIELD_CLEAR
OB50	1799	:	MAC	NOVECT	BAS	BASSFIELD_PURGE
OB50	1800	:	MAC	NOVECT	BAS	BASSFIELD_OPEN
OB50	1801	:	MAC	NOVECT	BAS	BASSFIELD_CLOSE
OB50	1802	:				
OB50	1803	:				
OB50	1804	:				
OB50	1805	:	MAC	NOVECT	BAS	BASSDATE T
OB50	1806	:	MAC	CALL	BAS	BASSERROR
OB58	1807	:	MAC	NOVECT	LIB	LIB\$DATE TIME
OB58	1808	:	MAC	NOVECT	BAS	BASSMARGIN
OB58	1809	:	MAC	NOVECT	BAS	BASSNOMARGIN
OB58	1810	:				
OB58	1811	:				
OB58	1812	:				
OB58	1813	:				
OB60	1814	:	MAC	CALL	LIB	LIB\$TPARSE
OB60	1815	:	MAC	NOVECT	LIB	LIB\$CVT_DTB
OB60	1815	:	MAC	NOVECT	LIB	LIB\$CVT-HTB
OB60	1816	:	MAC	NOVECT	LIB	LIB\$CVT-OTB

FIELD STATEMENT

MAC NOVECT BAS BASSFIELD_SET ; Set up FIELD var
MAC NOVECT BAS BASSFIELD_COPY ; Copy a FIELDed var
MAC NOVECT BAS BASSFIELD_CLEAR ; Clear the fielded attribu
MAC NOVECT BAS BASSFIELD_PURGE ; ?
MAC NOVECT BAS BASSFIELD_OPEN ; ?
MAC NOVECT BAS BASSFIELD_CLOSE ; ?

MISC

MAC NOVECT BAS BASSDATE T ; Return an ASCII string w/
MAC CALL BAS BASSERROR ; Signal errors from compil
MAC NOVECT LIB LIB\$DATE TIME ; System standard date/time
MAC NOVECT BAS BASSMARGIN ; MARGIN sta/MAR% funct
MAC NOVECT BAS BASSNOMARGIN ; NOMARGIN statement

LIB\$TPARSE and its subroutines

MAC CALL LIB LIB\$TPARSE
MAC NOVECT LIB LIB\$CVT_DTB
MAC NOVECT LIB LIB\$CVT-HTB
MAC NOVECT LIB LIB\$CVT-OTB

```

OB60 1818 :
OB60 1819 : Entry points used by the BASIC compiler to support the RUN command.
OB60 1820 :
OB60 1821 :
OB60 1822 : MAC NOVECT BAS BASSRUN_INIT ; Initialize for RUN
OB60 1823 : MAC CALL BAS BASSPUSH_ERR ; Save error status
OB68 1824 : MAC CALL BAS BASSPOP_ERR ; Restore error status
OB70 1825 : MAC NOVECT BAS BASSINIT_IOL ; Start immediate code

```

```
OB70 1826 :  
OB70 1827 : Internal BASIC entry points that are likely to need to be vectored  
OB70 1828 : because routines unlikely to be vectored call them.  
OB70 1829 :  
OB70 1830 : MAC JSB BAS BASS$CB_POP  
OB78 1831 : MAC JSB BAS BASS$CB_PUSH  
OB80 1832 : MAC JSB BAS BASS$CB_GET  
OB88 1833 : MAC CALL BAS BASS$ERR_INIT  
OB90 1834 : MAC CALL BAS BASS$OPEN_ZERO  
OB98 1835 : MAC CALL BAS BASS$RECO_INIT  
OBA0 1836 : MAC CALL BAS BASS$BLNK_LINE  
OBA8 1837 : MAC CALL BAS BASS$SIGNAL  
OBB0 1838 : MAC CALL BAS BASS$SIGNAL_IO  
OBB8 1839 : MAC CALL BAS BASS$STATU_INIT  
OBC0 1840 : MAC CALL BAS BASS$STOP  
OBC8 1841 : MAC CALL BAS BASS$STOP_IO  
OBD0 1842 : MAC CALL BAS BASS$CANTYP_AHEAD  
OBD8 1843 : MAC JSB BAS BASS$SCALE_L_R1  
OBE0 1844 : MAC JSB BAS BASS$SCALE_RT  
OBE8 1845 : MAC CALL BAS BASS$STOP_RMS  
OBF0 1846 : MAC CALL BAS BASS$FORMAT_INT  
OBF8 1847 : MAC CALL BAS BASS$CLOSE_ALL  
OC00 1848 : MAC CALL BAS BASS$UDF_R1  
OC08 1849 : MAC CALL BAS BASS$UDF_WL1  
OC10 1850 :  
OC10 1851 : JSB entry points to the string routines.  
OC10 1852 :  
OC10 1853 : MAC JSB STR STR$COPY_DX_R8  
OC18 1854 : MAC JSB STR STR$COPY_R_R8  
OC20 1855 : MAC JSB STR STR$DUPL_CHARR8  
OC28 1856 : MAC JSB STR STR$FREET_DX_R4  
OC30 1857 : MAC JSB STR STR$GET1_DX_R4  
OC38 1858 : MAC JSB STR STR$LEFT_R8  
OC40 1859 : MAC JSB STR STR$LEN_EXTR_R8  
OC48 1860 : MAC JSB STR STR$POSITION_R6  
OC50 1861 : MAC JSB STR STR$POS_EXTR_R8  
OC58 1862 : MAC JSB STR STR$RIGHT_R8  
OC60 1863 :  
OC60 1864 : More STR$ entry points. These modules must be in the sharable  
OC60 1865 : library, even though they are not used much, because they use  
OC60 1866 : string interlocks.  
OC60 1867 :  
OC60 1868 : MAC CALL STR STR$APPEND  
OC68 1869 : MAC CALL STR STR$COMPARE  
OC70 1870 : MAC CALL STR STR$COMPARE_EQ  
OC78 1871 : MAC CALL STR STR$PREFIX  
OC80 1872 : MAC CALL STR STR$REPLACE  
OC88 1873 : MAC JSB STR STR$REPLACE_R8  
OC90 1874 : MAC CALL STR STR$TRANSLATE  
OC98 1875 : MAC CALL STR STR$UPCASE
```

OCA0 1877 :
OCA0 1878 : The BASIC error codes. First the small integer symbols.
OCA0 1879 :
OCA0 1880 MAC SYM BAS BASSK_ACCDEVUSE
OCA0 1881 MAC SYM BAS BASSK_ARGDONMAT
OCA0 1882 MAC SYM BAS BASSK_ARGOUTBOU
OCA0 1883 MAC SYM BAS BASSK_ARGTOOLAR
OCA0 1884 MAC SYM BAS BASSK_ARRMUSSAM
OCA0 1885 MAC SYM BAS BASSK_ARRMUSSQU
OCA0 1886 MAC SYM BAS BASSK_BADDRDEV
OCA0 1887 MAC SYM BAS BASSK_BADLINNUM
OCA0 1888 MAC SYM BAS BASSK_BADNUMPRI
OCA0 1889 MAC SYM BAS BASSK_BADRECIDE
OCA0 1890 MAC SYM BAS BASSK_BADRECVAL
OCA0 1891 MAC SYM BAS BASSK_CANCHAARR
OCA0 1892 MAC SYM BAS BASSK_CANCOMSTA
OCA0 1893 MAC SYM BAS BASSK_CANCON
OCA0 1894 MAC SYM BAS BASSK_CANFINFIL
OCA0 1895 MAC SYM BAS BASSK_CANINVMAT
OCA0 1896 MAC SYM BAS BASSK_CANOPEFIL
OCA0 1897 MAC SYM BAS BASSK_CANPOSEOF
OCA0 1898 MAC SYM BAS BASSK_CHATO_NON
OCA0 1899 MAC SYM BAS BASSK_CORFICSTR
OCA0 1900 MAC SYM BAS BASSK_DATFORERR
OCA0 1901 MAC SYM BAS BASSK_DATTYPERR
OCA0 1902 MAC SYM BAS BASSK_DEFWITFNE
OCA0 1903 MAC SYM BAS BASSK_DEVHUNWRI
OCA0 1904 MAC SYM BAS BASSK_DEVNOTAVA
OCA0 1905 MAC SYM BAS BASSK_DEVNOTFIL
OCA0 1906 MAC SYM BAS BASSK_DIFUSELON
OCA0 1907 MAC SYM BAS BASSK_DIRERR
OCA0 1908 MAC SYM BAS BASSK_DISBLOINT
OCA0 1909 MAC SYM BAS BASSK_DISERRDUR
OCA0 1910 MAC SYM BAS BASSK_DISPACLOC
OCA0 1911 MAC SYM BAS BASSK_DISPACNEE
OCA0 1912 MAC SYM BAS BASSK_DISPACNOT
OCA0 1913 MAC SYM BAS BASSK_DISPACPRI
OCA0 1914 MAC SYM BAS BASSK_DIVBY_ZER
OCA0 1915 MAC SYM BAS BASSK_DUPKEYDET
OCA0 1916 MAC SYM BAS BASSK_ENDFILDEV
OCA0 1917 MAC SYM BAS BASSK_ERRTRANEE
OCA0 1918 MAC SYM BAS BASSK_ENDOF_STA
OCA0 1919 MAC SYM BAS BASSK_EXEONFIL
OCA0 1920 MAC SYM BAS BASSK_EXPERR
OCA0 1921 MAC SYM BAS BASSK_EXPTOOCOM
OCA0 1922 MAC SYM BAS BASSK_FATDISPAC
OCA0 1923 MAC SYM BAS BASSK_FATSYSIO
OCA0 1924 MAC SYM BAS BASSK_FIEOVEBUF
OCA0 1925 MAC SYM BAS BASSK_FILACPFAI
OCA0 1926 MAC SYM BAS BASSK_FILATTNOT
OCA0 1927 MAC SYM BAS BASSK_FILEXIREN
OCA0 1928 MAC SYM BAS BASSK_FILEXPDAT
OCA0 1929 MAC SYM BAS BASSK_FILIS_LOC
OCA0 1930 MAC SYM BAS BASSK_FIRARGSEQ
OCA0 1931 MAC SYM BAS BASSK_FLOOVE
OCA0 1932 MAC SYM BAS BASSK_FLOPOIERR
OCA0 1933 MAC SYM BAS BASSK_FLOUND

OCAO	1934	MAC	SYM	BAS	BASSK_FNEWITDEF
OCAO	1935	MAC	SYM	BAS	BASSK_FNEWITFUN
OCAO	1936	MAC	SYM	BAS	BASSK_FORWITNEX
OCAO	1937	MAC	SYM	BAS	BASSK_ILLALLCLA
OCAO	1938	MAC	SYM	BAS	BASSK_ILLARGLOG
OCAO	1939	MAC	SYM	BAS	BASSK_ILLBYTCOU
OCAO	1940	MAC	SYM	BAS	BASSK_ILLCLUSIZ
OCAO	1941	MAC	SYM	BAS	BASSK_ILLCONCLA
OCAO	1942	MAC	SYM	BAS	BASSK_ILLDEFNES
OCAO	1943	MAC	SYM	BAS	BASSK_ILLDUMVAI
OCAO	1944	MAC	SYM	BAS	BASSK_ILLEXIDEF
OCAO	1945	MAC	SYM	BAS	BASSK_ILLEXP
OCAO	1946	MAC	SYM	BAS	BASSK_ILLFIEVAR
OCAO	1947	MAC	SYM	BAS	BASSK_ILLFILNAM
OCAO	1948	MAC	SYM	BAS	BASSK_ILLFN RED
OCAO	1949	MAC	SYM	BAS	BASSK_ILLFUNNAM
OCAO	1950	MAC	SYM	BAS	BASSK_ILLIF STA
OCAO	1951	MAC	SYM	BAS	BASSK_ILLILACC
OCAO	1952	MAC	SYM	BAS	BASSK_ILLIN_IMM
OCAO	1953	MAC	SYM	BAS	BASSK_ILLIO_CHA
OCAO	1954	MAC	SYM	BAS	BASSK_ILLKEYATT
OCAO	1955	MAC	SYM	BAS	BASSK_ILLLINNUM
OCAO	1956	MAC	SYM	BAS	BASSK_ILLMAGUSA
OCAO	1957	MAC	SYM	BAS	BASSK_ILLMODMIX
OCAO	1958	MAC	SYM	BAS	BASSK_ILLNUM
OCAO	1959	MAC	SYM	BAS	BASSK_ILLNUMIMA
OCAO	1960	MAC	SYM	BAS	BASSK_ILLOPE
OCAO	1961	MAC	SYM	BAS	BASSK_ILLRECACC
OCAO	1962	MAC	SYM	BAS	BASSK_ILLRECFIL
OCAO	1963	MAC	SYM	BAS	BASSK_ILLRECFOR
OCAO	1964	MAC	SYM	BAS	BASSK_ILLRESSUB
OCAO	1965	MAC	SYM	BAS	BASSK_ILLRETSUB
OCAO	1966	MAC	SYM	BAS	BASSK_ILLSTA
OCAO	1967	MAC	SYM	BAS	BASSK_ILLSTRIMA
OCAO	1968	MAC	SYM	BAS	BASSK_ILLSWIUSA
OCAO	1969	MAC	SYM	BAS	BASSK_ILLSYM
OCAO	1970	MAC	SYM	BAS	BASSK_ILLSYSUSA
OCAO	1971	MAC	SYM	BAS	BASSK_ILLUSA
OCAO	1972	MAC	SYM	BAS	BASSK_ILLUSADEV
OCAO	1973	MAC	SYM	BAS	BASSK_ILLVER
OCAO	1974	MAC	SYM	BAS	BASSK_IMASQUROO
OCAO	1975	MAC	SYM	BAS	BASSK_INCFUNUSA
OCAO	1976	MAC	SYM	BAS	BASSK_INCSUBUSE
OCAO	1977	MAC	SYM	BAS	BASSK_INDNOTFUL
OCAO	1978	MAC	SYM	BAS	BASSK_INDNOTINI
OCAO	1979	MAC	SYM	BAS	BASSK_INTERR
OCAO	1980	MAC	SYM	BAS	BASSK_INTOVEFOR
OCAO	1981	MAC	SYM	BAS	BASSK_INVFILOPT
OCAO	1982	MAC	SYM	BAS	BASSK_INVKEYREF
OCAO	1983	MAC	SYM	BAS	BASSK_INVRFAFIE
OCAO	1984	MAC	SYM	BAS	BASSK_IO_CHAALR
OCAO	1985	MAC	SYM	BAS	BASSK_IO_CHANOT
OCAO	1986	MAC	SYM	BAS	BASSK_IO_TO DET
OCAO	1987	MAC	SYM	BAS	BASSK_KEYFIEBEY
OCAO	1988	MAC	SYM	BAS	BASSK_KEYLARTHA
OCAO	1989	MAC	SYM	BAS	BASSK_KEYNOTCHA
OCAO	1990	MAC	SYM	BAS	BASSK_KEYSIZTOO

[illegible]

OCA0	1991	MAC	SYM	BAS	BASSK_KEYWAIEXH
OCA0	1992	MAC	SYM	BAS	BASSK_LINTOOLON
OCA0	1993	MAC	SYM	BAS	BASSK_LITSTRNEE
OCA0	1994	MAC	SYM	BAS	BASSK_MAGRECLN
OCA0	1995	MAC	SYM	BAS	BASSK_MAGSELERR
OCA0	1996	MAC	SYM	BAS	BASSK_MATARRTOO
OCA0	1997	MAC	SYM	BAS	BASSK_MATARRWIT
OCA0	1998	MAC	SYM	BAS	BASSK_MATDIMERR
OCA0	1999	MAC	SYM	BAS	BASSK_MAXMEMEXC
OCA0	2000	MAC	SYM	BAS	BASSK_MEMMANVIO
OCA0	2001	MAC	SYM	BAS	BASSK_MEMPARFAI
OCA0	2002	MAC	SYM	BAS	BASSK_MISSPEFEA
OCA0	2003	MAC	SYM	BAS	BASSK_MODERR
OCA0	2004	MAC	SYM	BAS	BASSK_MOVOVEBUF
OCA0	2005	MAC	SYM	BAS	BASSK_NAMACCNOW
OCA0	2006	MAC	SYM	BAS	BASSK_NEGFILSTR
OCA0	2007	MAC	SYM	BAS	BASSK_NEXWITFOR
OCA0	2008	MAC	SYM	BAS	BASSK_NODNAMERR
OCA0	2009	MAC	SYM	BAS	BASSK_NONRESRUN
OCA0	2010	MAC	SYM	BAS	BASSK_NOTENDFIL
OCA0	2011	MAC	SYM	BAS	BASSK_NOTENOAVA
OCA0	2012	MAC	SYM	BAS	BASSK_NOTENODAT
OCA0	2013	MAC	SYM	BAS	BASSK_NOTIMP
OCA0	2014	MAC	SYM	BAS	BASSK_NOTRANACC
OCA0	2015	MAC	SYM	BAS	BASSK_NOTVALDEV
OCA0	2016	MAC	SYM	BAS	BASSK_NO_BUFSPA
OCA0	2017	MAC	SYM	BAS	BASSK_NO_CURREC
OCA0	2018	MAC	SYM	BAS	BASSK_NO_FIEIMA
OCA0	2019	MAC	SYM	BAS	BASSK_NO_FILNAM
OCA0	2020	MAC	SYM	BAS	BASSK_NO_PRIKEY
OCA0	2021	MAC	SYM	BAS	BASSK_NO_ROOUSE
OCA0	2022	MAC	SYM	BAS	BASSK_NO_RUNSYS
OCA0	2023	MAC	SYM	BAS	BASSK_NUCIMA
OCA0	2024	MAC	SYM	BAS	BASSK_NUMIMASTR
OCA0	2025	MAC	SYM	BAS	BASSK_NUMIS_NEE
OCA0	2026	MAC	SYM	BAS	BASSK_ODDADDTRA
OCA0	2027	MAC	SYM	BAS	BASSK_ONEOR_TWO
OCA0	2028	MAC	SYM	BAS	BASSK_ON_STANEE
OCA0	2029	MAC	SYM	BAS	BASSK_ON_STAOUT
OCA0	2030	MAC	SYM	BAS	BASSK_OUTOF_DAT
OCA0	2031	MAC	SYM	BAS	BASSK_PACIDSDON
OCA0	2032	MAC	SYM	BAS	BASSK_PLEUSERUN
OCA0	2033	MAC	SYM	BAS	BASSK_PRIKEYOUT
OCA0	2034	MAC	SYM	BAS	BASSK_PRIUSIBUF
OCA0	2035	MAC	SYM	BAS	BASSK_PRIUSIFOR
OCA0	2036	MAC	SYM	BAS	BASSK_PROC_TRA
OCA0	2037	MAC	SYM	BAS	BASSK_PROLOSSOR
OCA0	2038	MAC	SYM	BAS	BASSK_PROVIO
OCA0	2039	MAC	SYM	BAS	BASSK_RECALREXI
OCA0	2040	MAC	SYM	BAS	BASSK_RECATTNOT
OCA0	2041	MAC	SYM	BAS	BASSK_RECBUCLOC
OCA0	2042	MAC	SYM	BAS	BASSK_RECFILTOO
OCA0	2043	MAC	SYM	BAS	BASSK_RECHASBEE
OCA0	2044	MAC	SYM	BAS	BASSK_RECLOCFAI
OCA0	2045	MAC	SYM	BAS	BASSK_RECNOTFOU
OCA0	2046	MAC	SYM	BAS	BASSK_RECNUMEXC
OCA0	2047	MAC	SYM	BAS	BASSK_RECSUBCAL

OCA0	2048	MAC	SYM	BAS	BASSK_REDARR
OCA0	2049	MAC	SYM	BAS	BASSK_RESINSTRA
OCA0	2050	MAC	SYM	BAS	BASSK_RESNO_ERR
OCA0	2051	MAC	SYM	BAS	BASSK_RETWITGOS
OCA0	2052	MAC	SYM	BAS	BASSK_RRVNOTFUL
OCA0	2053	MAC	SYM	BAS	BASSK_SCAFACINT
OCA0	2054	MAC	SYM	BAS	BASSK_SIZRECINV
OCA0	2055	MAC	SYM	BAS	BASSK_SP_STAOVE
OCA0	2056	MAC	SYM	BAS	BASSK_STANOTFOU
OCA0	2057	MAC	SYM	BAS	BASSK_STO
OCA0	2058	MAC	SYM	BAS	BASSK_STRIMANUM
OCA0	2059	MAC	SYM	BAS	BASSK_STRIS_NEE
OCA0	2060	MAC	SYM	BAS	BASSK_STRTOOLON
OCA0	2061	MAC	SYM	BAS	BASSK_SUBOUTRAN
OCA0	2062	MAC	SYM	BAS	BASSK_SYNERR
OCA0	2063	MAC	SYM	BAS	BASSK_TAPBOTDET
OCA0	2064	MAC	SYM	BAS	BASSK_TAPNOTANS
OCA0	2065	MAC	SYM	BAS	BASSK_TAPRECNOT
OCA0	2066	MAC	SYM	BAS	BASSK_TERFORFIL
OCA0	2067	MAC	SYM	BAS	BASSK_TIMLIMEXC
OCA0	2068	MAC	SYM	BAS	BASSK_TOOFEWARG
OCA0	2069	MAC	SYM	BAS	BASSK_TOOMANARG
OCA0	2070	MAC	SYM	BAS	BASSK_TOOMANQPE
OCA0	2071	MAC	SYM	BAS	BASSK_UNDFUNCAL
OCA0	2072	MAC	SYM	BAS	BASSK_USEDATERR
OCA0	2073	MAC	SYM	BAS	BASSK_VIRARRDIS
OCA0	2074	MAC	SYM	BAS	BASSK_VIRARROPE
OCA0	2075	MAC	SYM	BAS	BASSK_VIRBUFTOO
OCA0	2076	MAC	SYM	BAS	BASSK_WHA
OCA0	2077	MAC	SYM	BAS	BASSK_WROMATPAC
OCA0	2078	:	:	:	:
OCA0	2079	:	:	:	:
OCA0	2080	:	:	:	:
OCA0	2081	MAC	SYM	BAS	BASSK_NEGZERTAB
OCA0	2082	MAC	SYM	BAS	BASSK_TOOMUCDAT
OCA0	2083	MAC	SYM	BAS	BASSK_ERRFILCOR
OCA0	2084	MAC	SYM	BAS	BASSK_UNEFILDAT
OCA0	2085	MAC	SYM	BAS	BASSK_NOSUPFOR
OCA0	2086	MAC	SYM	BAS	BASSK_DECERR
OCA0	2087	MAC	SYM	BAS	BASSK_NETOPEREJ
OCA0	2088	MAC	SYM	BAS	BASSK_REMOVEBUF
OCA0	2089	MAC	SYM	BAS	BASSK_UNAREMVAR
OCA0	2090	MAC	SYM	BAS	BASSK_RECOVEMAP
OCA0	2091	MAC	SYM	BAS	BASSK_IMPERRHAN
OCA0	2092	MAC	SYM	BAS	BASSK_ILLRECLOC
OCA0	2093	MAC	SYM	BAS	BASSK_REQRECSIZ
OCA0	2094	MAC	SYM	BAS	BASSK_TOOLITDAT
OCA0	2095	:	:	:	:
OCA0	2096	:	:	:	:
OCA0	2097	:	:	:	:
OCA0	2098	:	:	:	:
OCA0	2099	MAC	SYM	BAS	BASS_ACCDEVUSE
OCA0	2100	MAC	SYM	BAS	BASS_ARGDONMAT
OCA0	2101	MAC	SYM	BAS	BASS_ARGOUTBOU
OCA0	2102	MAC	SYM	BAS	BASS_ARGTOOLAR
OCA0	2103	MAC	SYM	BAS	BASS_ARRMUSSAM
OCA0	2104	MAC	SYM	BAS	BASS_ARRMUSSQU

OCA0	2105	MAC	SYM	BAS	BASS_BADDIRDEV
OCA0	2106	MAC	SYM	BAS	BASS_BADLINNUM
OCA0	2107	MAC	SYM	BAS	BASS_BADNUMPRI
OCA0	2108	MAC	SYM	BAS	BASS_BADRECIDE
OCA0	2109	MAC	SYM	BAS	BASS_BADRECVAL
OCA0	2110	MAC	SYM	BAS	BASS_CANCHAARR
OCA0	2111	MAC	SYM	BAS	BASS_CANCOMSTA
OCA0	2112	MAC	SYM	BAS	BASS_CANCON
OCA0	2113	MAC	SYM	BAS	BASS_CANFINFIL
OCA0	2114	MAC	SYM	BAS	BASS_CANINVMAT
OCA0	2115	MAC	SYM	BAS	BASS_CANOPEFIL
OCA0	2116	MAC	SYM	BAS	BASS_CANPOSEOF
OCA0	2117	MAC	SYM	BAS	BASS_CHATO_NON
OCA0	2118	MAC	SYM	BAS	BASS_CORFILSTR
OCA0	2119	MAC	SYM	BAS	BASS_DATFORERR
OCA0	2120	MAC	SYM	BAS	BASS_DATTYPERR
OCA0	2121	MAC	SYM	BAS	BASS_DEFWITFNE
OCA0	2122	MAC	SYM	BAS	BASS_DEVHUNWRI
OCA0	2123	MAC	SYM	BAS	BASS_DEVNOTAVA
OCA0	2124	MAC	SYM	BAS	BASS_DEVNOTFIL
OCA0	2125	MAC	SYM	BAS	BASS_DIFUSELON
OCA0	2126	MAC	SYM	BAS	BASS_DIRERR
OCA0	2127	MAC	SYM	BAS	BASS_DISBLOINT
OCA0	2128	MAC	SYM	BAS	BASS_DISERRDUR
OCA0	2129	MAC	SYM	BAS	BASS_DISPACLOC
OCA0	2130	MAC	SYM	BAS	BASS_DISPACNEE
OCA0	2131	MAC	SYM	BAS	BASS_DISPACNOT
OCA0	2132	MAC	SYM	BAS	BASS_DISPACPRI
OCA0	2133	MAC	SYM	BAS	BASS_DIVBY_ZER
OCA0	2134	MAC	SYM	BAS	BASS_DUPKEYDET
OCA0	2135	MAC	SYM	BAS	BASS_ENDFILDEV
OCA0	2136	MAC	SYM	BAS	BASS_ENDOF_STA
OCA0	2137	MAC	SYM	BAS	BASS_ERRTRANE
OCA0	2138	MAC	SYM	BAS	BASS_EXEONLFIL
OCA0	2139	MAC	SYM	BAS	BASS_EXPERR
OCA0	2140	MAC	SYM	BAS	BASS_EXPTOOCOM
OCA0	2141	MAC	SYM	BAS	BASS_FATDISPAC
OCA0	2142	MAC	SYM	BAS	BASS_FATSYSIO
OCA0	2143	MAC	SYM	BAS	BASS_FIEOVEBUF
OCA0	2144	MAC	SYM	BAS	BASS_FILACPFAI
OCA0	2145	MAC	SYM	BAS	BASS_FILATTNOT
OCA0	2146	MAC	SYM	BAS	BASS_FILEXIREN
OCA0	2147	MAC	SYM	BAS	BASS_FILEXPDAT
OCA0	2148	MAC	SYM	BAS	BASS_FILIS_LOC
OCA0	2149	MAC	SYM	BAS	BASS_FIRARGSEQ
OCA0	2150	MAC	SYM	BAS	BASS_FLOOVE
OCA0	2151	MAC	SYM	BAS	BASS_FLOPOIERR
OCA0	2152	MAC	SYM	BAS	BASS_FLOUND
OCA0	2153	MAC	SYM	BAS	BASS_FNEWITDEF
OCA0	2154	MAC	SYM	BAS	BASS_FNEWITFUN
OCA0	2155	MAC	SYM	BAS	BASS_FORWITNEX
OCA0	2156	MAC	SYM	BAS	BASS_ILLALLCLA
OCA0	2157	MAC	SYM	BAS	BASS_ILLARGLOG
OCA0	2158	MAC	SYM	BAS	BASS_ILLBYTCOU
OCA0	2159	MAC	SYM	BAS	BASS_ILLCLUSIZ
OCA0	2160	MAC	SYM	BAS	BASS_ILLCONCLA
OCA0	2161	MAC	SYM	BAS	BASS_ILLDEFNES

OCA0	2162	MAC	SYM	BAS	BASS-ILLDUMVAR
OCA0	2163	MAC	SYM	BAS	BASS-ILLEXIDF
OCA0	2164	MAC	SYM	BAS	BASS-ILLEX
OCA0	2165	MAC	SYM	BAS	BASS-ILLFIEVAR
OCA0	2166	MAC	SYM	BAS	BASS-ILLFILNAM
OCA0	2167	MAC	SYM	BAS	BASS-ILLFN RED
OCA0	2168	MAC	SYM	BAS	BASS-ILLFUNNAM
OCA0	2169	MAC	SYM	BAS	BASS-ILLIF STA
OCA0	2170	MAC	SYM	BAS	BASS-ILLILCACC
OCA0	2171	MAC	SYM	BAS	BASS-ILLIN IMM
OCA0	2172	MAC	SYM	BAS	BASS-ILLIO CHA
OCA0	2173	MAC	SYM	BAS	BASS-ILLKEYATT
OCA0	2174	MAC	SYM	BAS	BASS-ILLLINNUM
OCA0	2175	MAC	SYM	BAS	BASS-ILLMAGUSA
OCA0	2176	MAC	SYM	BAS	BASS-ILLMODMIX
OCA0	2177	MAC	SYM	BAS	BASS-ILLNUM
OCA0	2178	MAC	SYM	BAS	BASS-ILLNUMIMA
OCA0	2179	MAC	SYM	BAS	BASS-ILLOPE
OCA0	2180	MAC	SYM	BAS	BASS-ILLRECACC
OCA0	2181	MAC	SYM	BAS	BASS-ILLRECFIL
OCA0	2182	MAC	SYM	BAS	BASS-ILLRECFOR
OCA0	2183	MAC	SYM	BAS	BASS-ILLRESSUB
OCA0	2184	MAC	SYM	BAS	BASS-ILLRETSUB
OCA0	2185	MAC	SYM	BAS	BASS-ILLSTA
OCA0	2186	MAC	SYM	BAS	BASS-ILLSTRIMA
OCA0	2187	MAC	SYM	BAS	BASS-ILLSWIUSA
OCA0	2188	MAC	SYM	BAS	BASS-ILLSYM
OCA0	2189	MAC	SYM	BAS	BASS-ILLSYSUSA
OCA0	2190	MAC	SYM	BAS	BASS-ILLUSA
OCA0	2191	MAC	SYM	BAS	BASS-ILLUSADEV
OCA0	2192	MAC	SYM	BAS	BASS-ILLVER
OCA0	2193	MAC	SYM	BAS	BASS-IMASQUROO
OCA0	2194	MAC	SYM	BAS	BASS-INCFUNUSA
OCA0	2195	MAC	SYM	BAS	BASS-INCSUBUSE
OCA0	2196	MAC	SYM	BAS	BASS-INDNOTFUL
OCA0	2197	MAC	SYM	BAS	BASS-INDNOTINI
OCA0	2198	MAC	SYM	BAS	BASS-INTERR
OCA0	2199	MAC	SYM	BAS	BASS-INTOVEFOR
OCA0	2200	MAC	SYM	BAS	BASS-INVFILOPT
OCA0	2201	MAC	SYM	BAS	BASS-INVKEYREF
OCA0	2202	MAC	SYM	BAS	BASS-INVRFIE
OCA0	2203	MAC	SYM	BAS	BASS-IO CHAALR
OCA0	2204	MAC	SYM	BAS	BASS-IO CHANOT
OCA0	2205	MAC	SYM	BAS	BASS-IO TO DET
OCA0	2206	MAC	SYM	BAS	BASS-KEYFIEBEY
OCA0	2207	MAC	SYM	BAS	BASS-KEYLARTHA
OCA0	2208	MAC	SYM	BAS	BASS-KEYNOTCHA
OCA0	2209	MAC	SYM	BAS	BASS-KEYSIZTOO
OCA0	2210	MAC	SYM	BAS	BASS-KEYWAIEXH
OCA0	2211	MAC	SYM	BAS	BASS-LINTOOLON
OCA0	2212	MAC	SYM	BAS	BASS-LITSTRNEE
OCA0	2213	MAC	SYM	BAS	BASS-MAGRECLN
OCA0	2214	MAC	SYM	BAS	BASS-MAGSELERR
OCA0	2215	MAC	SYM	BAS	BASS-MATARRTOO
OCA0	2216	MAC	SYM	BAS	BASS-MATARRWIT
OCA0	2217	MAC	SYM	BAS	BASS-MATDIMERR
OCA0	2218	MAC	SYM	BAS	BASS-MAXMEMEXC

OCA0	2219	MAC	SYM	BAS	BASS-MEMMANVIO
OCA0	2220	MAC	SYM	BAS	BASS-MEMPARFAI
OCA0	2221	MAC	SYM	BAS	BASS-MISSPEFEA
OCA0	2222	MAC	SYM	BAS	BASS-MODERR
OCA0	2223	MAC	SYM	BAS	BASS-MOVOVEBUF
OCA0	2224	MAC	SYM	BAS	BASS-NAMACCNOW
OCA0	2225	MAC	SYM	BAS	BASS-NEGFILSTR
OCA0	2226	MAC	SYM	BAS	BASS-NEXWITFOR
OCA0	2227	MAC	SYM	BAS	BASS-NODNAMERR
OCA0	2228	MAC	SYM	BAS	BASS-NONRESRUN
OCA0	2229	MAC	SYM	BAS	BASS-NOTENDFIL
OCA0	2230	MAC	SYM	BAS	BASS-NOTENOAVA
OCA0	2231	MAC	SYM	BAS	BASS-NOTENODAT
OCA0	2232	MAC	SYM	BAS	BASS-NOTIMP
OCA0	2233	MAC	SYM	BAS	BASS-NOTRANACC
OCA0	2234	MAC	SYM	BAS	BASS-NOTVALDEV
OCA0	2235	MAC	SYM	BAS	BASS-NO_BUFSPA
OCA0	2236	MAC	SYM	BAS	BASS-NO_CURREC
OCA0	2237	MAC	SYM	BAS	BASS-NO_FIEIMA
OCA0	2238	MAC	SYM	BAS	BASS-NO_FILNAM
OCA0	2239	MAC	SYM	BAS	BASS-NO_PRIKEY
OCA0	2240	MAC	SYM	BAS	BASS-NO_ROOUSE
OCA0	2241	MAC	SYM	BAS	BASS-NO_RUNSYS
OCA0	2242	MAC	SYM	BAS	BASS-NUCIMA
OCA0	2243	MAC	SYM	BAS	BASS-NUMIMASTR
OCA0	2244	MAC	SYM	BAS	BASS-NUMIS_NEE
OCA0	2245	MAC	SYM	BAS	BASS-ODDADDTRA
OCA0	2246	MAC	SYM	BAS	BASS-ONEOR_TWO
OCA0	2247	MAC	SYM	BAS	BASS-ON_STA_NEE
OCA0	2248	MAC	SYM	BAS	BASS-ON_STAOUT
OCA0	2249	MAC	SYM	BAS	BASS-OUTOF_DAT
OCA0	2250	MAC	SYM	BAS	BASS-PACIDSDON
OCA0	2251	MAC	SYM	BAS	BASS-PLUSERUN
OCA0	2252	MAC	SYM	BAS	BASS-PRIKEYOUT
OCA0	2253	MAC	SYM	BAS	BASS-PRIUSIBUF
OCA0	2254	MAC	SYM	BAS	BASS-PRIUSIFOR
OCA0	2255	MAC	SYM	BAS	BASS-PROC_TRA
OCA0	2256	MAC	SYM	BAS	BASS-PROLOSSOR
OCA0	2257	MAC	SYM	BAS	BASS-PROVIO
OCA0	2258	MAC	SYM	BAS	BASS-RECALREXI
OCA0	2259	MAC	SYM	BAS	BASS-RECATTNOT
OCA0	2260	MAC	SYM	BAS	BASS-RECBUCLOC
OCA0	2261	MAC	SYM	BAS	BASS-RECFILTOO
OCA0	2262	MAC	SYM	BAS	BASS-RECHASBEE
OCA0	2263	MAC	SYM	BAS	BASS-RECLOCFAI
OCA0	2264	MAC	SYM	BAS	BASS-RECNOTFOU
OCA0	2265	MAC	SYM	BAS	BASS-RECNUMEXC
OCA0	2266	MAC	SYM	BAS	BASS-RECSUBCAL
OCA0	2267	MAC	SYM	BAS	BASS-REDARR
OCA0	2268	MAC	SYM	BAS	BASS-RESINSTRA
OCA0	2269	MAC	SYM	BAS	BASS-RESNO_ERR
OCA0	2270	MAC	SYM	BAS	BASS-RETWITGOS
OCA0	2271	MAC	SYM	BAS	BASS-RRVNOTFUL
OCA0	2272	MAC	SYM	BAS	BASS-SCAFACINT
OCA0	2273	MAC	SYM	BAS	BASS-SIZRECINV
OCA0	2274	MAC	SYM	BAS	BASS-SP_STA_OVE
OCA0	2275	MAC	SYM	BAS	BASS-STANOTFOU

OCA0	2276	MAC	SYM	BAS	BASS_STO
OCA0	2277	MAC	SYM	BAS	BASS_STRIMANUM
OCA0	2278	MAC	SYM	BAS	BASS_STRIS_NEE
OCA0	2279	MAC	SYM	BAS	BASS_STRTODLON
OCA0	2280	MAC	SYM	BAS	BASS_SUBOUTRAN
OCA0	2281	MAC	SYM	BAS	BASS_SYNERR
OCA0	2282	MAC	SYM	BAS	BASS_TAPBOTDET
OCA0	2283	MAC	SYM	BAS	BASS_TAPNOTANS
OCA0	2284	MAC	SYM	BAS	BASS_TAPRECNOT
OCA0	2285	MAC	SYM	BAS	BASS_TERFORFIL
OCA0	2286	MAC	SYM	BAS	BASS_TIMLIMEXC
OCA0	2287	MAC	SYM	BAS	BASS_TOOFEWARG
OCA0	2288	MAC	SYM	BAS	BASS_TOOMANARG
OCA0	2289	MAC	SYM	BAS	BASS_TOOMANOPE
OCA0	2290	MAC	SYM	BAS	BASS_UNDFUNCAL
OCA0	2291	MAC	SYM	BAS	BASS_USEDATERR
OCA0	2292	MAC	SYM	BAS	BASS_VIRARRDIS
OCA0	2293	MAC	SYM	BAS	BASS_VIRARROPE
OCA0	2294	MAC	SYM	BAS	BASS_VIRBUFTOO
OCA0	2295	MAC	SYM	BAS	BASS_WHA
OCA0	2296	MAC	SYM	BAS	BASS_WROMATPAC
OCA0	2297				
OCA0	2298	; New messages for Basic 2.0, VMS 3.1			
OCA0	2299	MAC	SYM	BAS	BASS_NEGZERTAB
OCA0	2300	MAC	SYM	BAS	BASS_TOOMUCDAT
OCA0	2301	MAC	SYM	BAS	BASS_ERRFILCOR
OCA0	2302	MAC	SYM	BAS	BASS_UNEFILDAT
OCA0	2303	MAC	SYM	BAS	BASS_NOSUPFOR
OCA0	2304	MAC	SYM	BAS	BASS_DECERR
OCA0	2305	MAC	SYM	BAS	BASS_NETOPEREJ
OCA0	2306	MAC	SYM	BAS	BASS_REMOVEBUF
OCA0	2307	MAC	SYM	BAS	BASS_UNAREMVAR
OCA0	2308	MAC	SYM	BAS	BASS_RECOVEMAP
OCA0	2309	MAC	SYM	BAS	BASS_IMPERRHAN
OCA0	2310	MAC	SYM	BAS	BASS_ILLRECLOC
OCA0	2311	MAC	SYM	BAS	BASS_REQRECSIZ
OCA0	2312	MAC	SYM	BAS	BASS_TOOLITDAT
OCA0	2313				
OCA0	2314	; Module BASS\$REC_PROC			
OCA0	2315	MAC	CALL	BAS	BASSWAIT

```

OCA8 2317 :+
OCA8 2318 : START OF COBOL CONTRIBUTION TO RTLVECTOR
OCA8 2319 :
OCA8 2320 : Leave some expansion room for BAS by padding to a page boundary.
OCA8 2321 : If this room is exhausted they will have to go after
OCA8 2322 : the COB facility. Give an error message if the space gets
OCA8 2323 : exhausted.
OCA8 2324 :-
FFFFFEAB OCA8 2325 .IF GREATER <<.-RTL$START>--^XE00>
OCA8 2326 .ERROR <<.-RTL$START>--^XE00> ; Negative vector pad space
OCA8 2327 .ENDC
'00'00'00' OCA8 2328 .BYTE 0[512-<<.-RTL$START>&511>]

```

00'00'00'00'00'00'00'00'00'00'00'00'	0CA8
00'00'00'00'00'00'00'00'00'00'00'00'	0CB8
00'00'00'00'00'00'00'00'00'00'00'00'	0CB4
00'00'00'00'00'00'00'00'00'00'00'00'	0CC0
00'00'00'00'00'00'00'00'00'00'00'00'	0CCC
00'00'00'00'00'00'00'00'00'00'00'00'	0CD8
00'00'00'00'00'00'00'00'00'00'00'00'	0CE4
00'00'00'00'00'00'00'00'00'00'00'00'	0CF0
00'00'00'00'00'00'00'00'00'00'00'00'	0CFC
00'00'00'00'00'00'00'00'00'00'00'00'	0D08
00'00'00'00'00'00'00'00'00'00'00'00'	0D14
00'00'00'00'00'00'00'00'00'00'00'00'	0D20
00'00'00'00'00'00'00'00'00'00'00'00'	0D2C
00'00'00'00'00'00'00'00'00'00'00'00'	0D38
00'00'00'00'00'00'00'00'00'00'00'00'	0D44
00'00'00'00'00'00'00'00'00'00'00'00'	0D50
00'00'00'00'00'00'00'00'00'00'00'00'	0D5C
00'00'00'00'00'00'00'00'00'00'00'00'	0D68
00'00'00'00'00'00'00'00'00'00'00'00'	0D74
00'00'00'00'00'00'00'00'00'00'00'00'	0D80
00'00'00'00'00'00'00'00'00'00'00'00'	0D8C
00'00'00'00'00'00'00'00'00'00'00'00'	0D98
00'00'00'00'00'00'00'00'00'00'00'00'	0DA4
00'00'00'00'00'00'00'00'00'00'00'00'	0DB0
00'00'00'00'00'00'00'00'00'00'00'00'	0DBC
00'00'00'00'00'00'00'00'00'00'00'00'	0DC8
00'00'00'00'00'00'00'00'00'00'00'00'	0DD4
00'00'00'00'00'00'00'00'00'00'00'00'	0DE0
00'00'00'00'00'00'00'00'00'00'00'00'	0DEC
00'00'00'00'00'00'00'00'00'00'00'00'	0DF8

```
OE00 2330 ; MODULE LIB$AB_CVTPT_0
OE00 2331      MAC      SYM      LIB      LIB$AB_CVTPT_0
OE00 2332
OE00 2333 ; MODULE LIB$AB_CVTPT_U
OE00 2334      MAC      SYM      LIB      LIB$AB_CVTPT_U
OE00 2335
OE00 2336 ; MODULE LIB$AB_CVTTP_0
OE00 2337      MAC      SYM      LIB      LIB$AB_CVTTP_0
OE00 2338
OE00 2339 ; MODULE LIB$AB_CVTTP_U
OE00 2340      MAC      SYM      LIB      LIB$AB_CVTTP_U
OE00 2341
OE00 2342 ; MODULE COB$AB_SPANC
OE00 2343      MAC      SYM      COB      COB$AB_SPANC
OE00 2344
OE00 2345 ; MODULE LIB$AB_CVT_U_0
OE00 2346      MAC      SYM      LIB      LIB$AB_CVT_U_0
OE00 2347
OE00 2348 ;+
OE00 2349 ; Degree equivalents of trig functions
OE00 2350 ;-
OE00 2351
OE00 2352 ; MODULE:MTH$ACOS
OE00 2353      MAC      CALL      MTH      MTH$ACOSD
OE08 2354      MAC      JSB      MTH      MTH$ACOSD_R4
OE10 2355
OE10 2356 ; MODULE:MTH$ASIN
OE10 2357      MAC      CALL      MTH      MTH$ASIND
OE18 2358      MAC      JSB      MTH      MTH$ASIND_R4
OE20 2359
OE20 2360 ; MODULE:MTH$ATAN
OE20 2361      MAC      CALL      MTH      MTH$ATAND
OE28 2362      MAC      CALL      MTH      MTH$ATAND2
OE30 2363      MAC      JSB      MTH      MTH$ATAND_R4
OE38 2364
OE38 2365 ; MODULE:MTH$DACOS
OE38 2366      MAC      CALL      MTH      MTH$DACOSD
OE40 2367      MAC      JSB      MTH      MTH$DACOSD_R7
OE48 2368
OE48 2369 ; MODULE:MTH$DASIN
OE48 2370      MAC      CALL      MTH      MTH$DASIND
OE50 2371      MAC      JSB      MTH      MTH$DASIND_R7
OE58 2372
OE58 2373 ; MODULE:MTH$DATAN
OE58 2374      MAC      CALL      MTH      MTH$DATAND
OE60 2375      MAC      CALL      MTH      MTH$DATAND2
OE68 2376      MAC      JSB      MTH      MTH$DATAND_R7
OE70 2377
OE70 2378 ; MODULE:MTH$DSINCOS
OE70 2379      MAC      CALL      MTH      MTH$DCOSD
OE78 2380      MAC      JSB      MTH      MTH$DCOSD_R7
OE80 2381      MAC      CALL      MTH      MTH$DSIND
OE88 2382      MAC      JSB      MTH      MTH$DSIND_R7
OE90 2383
OE90 2384 ; MODULE:MTH$SINCOS
OE90 2385      MAC      CALL      MTH      MTH$SCOSD
OE98 2386      MAC      JSB      MTH      MTH$SCOSD_R4
```

```
OEA0 2387      MAC      CALL      MTH      MTH$SIND
OEA8 2388      MAC      JSB      MTH      MTH$SIND_R4
OEB0 2389
OEB0 2390 ; MODULE:MTH$DTAN
OEB0 2391      MAC      CALL      MTH      MTH$DTAND
OEB8 2392      MAC      JSB      MTH      MTH$DTAND_R7
OEC0 2393
OEC0 2394 ; MODULE:MTH$STAN
OEC0 2395      MAC      CALL      MTH      MTH$STAND
OEC8 2396      MAC      JSB      MTH      MTH$STAND_R4
OED0 2397
OED0 2398 ; MODULE MTH$GACOS
OED0 2399      MAC      NOVECT    MTH      MTH$GACOSD
OED0 2400      MAC      NOVECT    MTH      MTH$GACOSD_R7
OED0 2401
OED0 2402 ; MODULE MTH$GASIN
OED0 2403      MAC      NOVECT    MTH      MTH$GASIND
OED0 2404      MAC      NOVECT    MTH      MTH$GASIND_R7
OED0 2405
OED0 2406 ; MODULE MTH$GSINCOS
OED0 2407      MAC      NOVECT    MTH      MTH$GSIND
OED0 2408      MAC      NOVECT    MTH      MTH$GCOSD
OED0 2409      MAC      NOVECT    MTH      MTH$GSIND_R7
OED0 2410      MAC      NOVECT    MTH      MTH$GCOSD_R7
OED0 2411
OED0 2412 ; MODULE MTH$GTAN
OED0 2413      MAC      NOVECT    MTH      MTH$GTAND
OED0 2414      MAC      NOVECT    MTH      MTH$GTAND_R7
OED0 2415
OED0 2416 ; MODULE MTH$HACOS
OED0 2417      MAC      NOVECT    MTH      MTH$HACOSD
OED0 2418      MAC      NOVECT    MTH      MTH$HACOSD_R8
OED0 2419
OED0 2420 ; MODULE MTH$HASIN
OED0 2421      MAC      NOVECT    MTH      MTH$HASIND
OED0 2422      MAC      NOVECT    MTH      MTH$HASIND_R8
OED0 2423
OED0 2424 ; MODULE MTH$HATANH
OED0 2425      MAC      NOVECT    MTH      MTH$HATANH
OED0 2426
OED0 2427 ; MODULE MTH$HSINCOS
OED0 2428      MAC      NOVECT    MTH      MTH$HSIND
OED0 2429      MAC      NOVECT    MTH      MTH$HSIND_R5
OED0 2430      MAC      NOVECT    MTH      MTH$HCOSD
OED0 2431      MAC      NOVECT    MTH      MTH$HCOSD_R5
OED0 2432
OED0 2433 ; MODULE MTH$HTAN
OED0 2434      MAC      NOVECT    MTH      MTH$HTAND
OED0 2435      MAC      NOVECT    MTH      MTH$HTAND_R5
OED0 2436
OEDC 2437 ; V E R S I O N      3 . 0      A D D I T I O N S
OED0 2438
OED0 2439 ; FITT up the hole created by removing some translation tables that ended up
OED0 2440 ; in the vector.
OED0 2441
OED0 2442 ; MODULE:MTH$ATANH
OED0 2443      MAC      CALL      MTH      MTH$ATANH
```

```
OED8 2444
OED8 2445 ; MODULE:MTH$DATANH
OED8 2446     MAC      CALL      MTH      MTH$DATANH
OEE0 2447
OEE0 2448 ; MODULE MTH$GATAN
OEE0 2449     MAC      NOVECT     MTH      MTH$GATAN
OEE0 2450     MAC      NOVECT     MTH      MTH$GATAN2
OEE0 2451     MAC      NOVECT     MTH      MTH$GATAN_R7
OEE0 2452     MAC      NOVECT     MTH      MTH$GATAN_D
OEE0 2453     MAC      NOVECT     MTH      MTH$GATAN_D2
OEE0 2454     MAC      NOVECT     MTH      MTH$GATAN_D_R7
OEE0 2455
OEE0 2456 ; MODULE MTH$GLOG
OEE0 2457     MAC      NOVECT     MTH      MTH$GLOG
OEE0 2458     MAC      NOVECT     MTH      MTH$GLOG2
OEE0 2459     MAC      NOVECT     MTH      MTH$GLOG10
OEE0 2460     MAC      NOVECT     MTH      MTH$GLOG_R8
OEE0 2461     MAC      NOVECT     MTH      MTH$GLOG10_R8
OEE0 2462
OEE0 2463 ; MODULE MTH$HATAN
OEE0 2464     MAC      NOVECT     MTH      MTH$HATAN
OEE0 2465     MAC      NOVECT     MTH      MTH$HATAN_R8
OEE0 2466     MAC      NOVECT     MTH      MTH$HATAN2
OEE0 2467     MAC      NOVECT     MTH      MTH$HATAN_D
OEE0 2468     MAC      NOVECT     MTH      MTH$HATAN_D_R8
OEE0 2469     MAC      NOVECT     MTH      MTH$HATAN_D2
OEE0 2470
OEE0 2471 ; MODULE MTH$HLOG
OEE0 2472     MAC      NOVECT     MTH      MTH$HLOG
OEE0 2473     MAC      NOVECT     MTH      MTH$HLOG2
OEE0 2474     MAC      NOVECT     MTH      MTH$HLOG10
OEE0 2475     MAC      NOVECT     MTH      MTH$HLOG_R8
OEE0 2476     MAC      NOVECT     MTH      MTH$HLOG10_R8
OEE0 2477
OEE0 2478 ; MODULE MTH$SINCOS      (Continued)
OEE0 2479     MAC      CALL      MTH      MTH$SINCOS
OEE8 2480     MAC      JSB       MTH      MTH$SINCOS_R5
OEF0 2481     MAC      CALL      MTH      MTH$SINCOS_D
OEF8 2482     MAC      JSB       MTH      MTH$SINCOS_D_R5
OF00 2483
OF00 2484 ; MODULE MTH$DSINCOS      (Continued)
OF00 2485     MAC      CALL      MTH      MTH$DSINCOS
OF08 2486     MAC      JSB       MTH      MTH$DSINCOS_R7
OF10 2487     MAC      CALL      MTH      MTH$DSINCOS_D
OF18 2488     MAC      JSB       MTH      MTH$DSINCOS_D_R7
OF20 2489
OF20 2490 ; MODULE MTH$GSINCOS      (Continued)
OF20 2491     MAC      NOVECT     MTH      MTH$GSINCOS
OF20 2492     MAC      NOVECT     MTH      MTH$GSINCOS_R7
OF20 2493     MAC      NOVECT     MTH      MTH$GSINCOS_D
OF20 2494     MAC      NOVECT     MTH      MTH$GSINCOS_D_R7
OF20 2495
OF20 2496 ; MODULE MTH$HSINCOS      (Continued)
OF20 2497     MAC      NOVECT     MTH      MTH$HSINCOS
OF20 2498     MAC      NOVECT     MTH      MTH$HSINCOS_R7
OF20 2499     MAC      NOVECT     MTH      MTH$HSINCOS_D
OF20 2500     MAC      NOVECT     MTH      MTH$HSINCOS_D_R7
```

OF20	2501				
OF20	2502	; MODULE:MTH\$ALOG	(Continued)		
OF20	2503	MAC CALL	MTH	MTH\$ALOG2	
OF28	2504				
OF28	2505	; MODULE:MTH\$DLOG	(Continued)		
OF28	2506	MAC CALL	MTH	MTH\$DLOG2	
OF30	2507				
OF30	2508	; MODULE MTH\$AL_4_OV_PI			
OF30	2509	MAC NOVECT	MTH	MTH\$AL_4_OV_PI	
OF30	2510				
OF30	2511	; MODULE MTH\$TAN	(Continued)		
OF30	2512	MAC JSB	MTH	MTH\$TAN_R5	
OF38	2513	MAC JSB	MTH	MTH\$TAN_R5	
OF40	2514				
OF40	2515	; MODULE MTH\$HTAN	(Continued)		
OF40	2516	MAC NOVECT	MTH	MTH\$HTAN_R7	
OF40	2517	MAC NOVECT	MTH	MTH\$HTAN_R7	
OF40	2518				
OF40	2519	; MODULE MTH\$AL_4_OV_PI			
OF40	2520	MAC DATA	MTH	MTH\$AL_4_OV_PI	
OF48	2521				
OF48	2522	; MODULE MTH\$ALOG			
OF48	2523	MAC DATA	MTH	MTH\$\$AB ALOG	
OF50	2524				
OF50	2525	; MODULE MTH\$ATAN			
OF50	2526	MAC DATA	MTH	MTH\$\$AB ATAN	

	OF58	2528	:	+
	OF58	2529	.	
	OF58	2530	.	
	OF58	2531	.	
	OF58	2532	.	
	OF58	2533	.	
	OF58	2534	.	-
FFFFFC40	OF58	2535		
	OF58	2536		
	OF58	2537		
	OF58	2538		
00'00'00'00'00'00'00'00'00'00'00'00'	OF58			
00'00'00'00'00'00'00'00'00'00'00'00'	OF64			
00'00'00'00'00'00'00'00'00'00'00'00'	OF70			
00'00'00'00'00'00'00'00'00'00'00'00'	OF7C			
00'00'00'00'00'00'00'00'00'00'00'00'	OF88			
00'00'00'00'00'00'00'00'00'00'00'00'	OF94			
00'00'00'00'00'00'00'00'00'00'00'00'	0FA0			
00'00'00'00'00'00'00'00'00'00'00'00'	0FAC			
00'00'00'00'00'00'00'00'00'00'00'00'	0FB8			
00'00'00'00'00'00'00'00'00'00'00'00'	0FC4			
00'00'00'00'00'00'00'00'00'00'00'00'	0FD0			
00'00'00'00'00'00'00'00'00'00'00'00'	0FDC			
00'00'00'00'00'00'00'00'00'00'00'00'	0FE8			
00'00'00'00'00'00'00'00'00'00'00'00'	0FF4			
00'00'00'00'00'00'00'00'00'00'00'00'	1000			
00'00'00'00'00'00'00'00'00'00'00'00'	100C			
00'00'00'00'00'00'00'00'00'00'00'00'	1018			
00'00'00'00'00'00'00'00'00'00'00'00'	1024			
00'00'00'00'00'00'00'00'00'00'00'00'	1030			
00'00'00'00'00'00'00'00'00'00'00'00'	103C			
00'00'00'00'00'00'00'00'00'00'00'00'	1048			
00'00'00'00'00'00'00'00'00'00'00'00'	1054			
00'00'00'00'00'00'00'00'00'00'00'00'	1060			
00'00'00'00'00'00'00'00'00'00'00'00'	106C			
00'00'00'00'00'00'00'00'00'00'00'00'	1078			
00'00'00'00'00'00'00'00'00'00'00'00'	1084			
00'00'00'00'00'00'00'00'00'00'00'00'	1090			
00'00'00'00'00'00'00'00'00'00'00'00'	109C			
00'00'00'00'00'00'00'00'00'00'00'00'	10A8			
00'00'00'00'00'00'00'00'00'00'00'00'	10B4			
00'00'00'00'00'00'00'00'00'00'00'00'	10C0			
00'00'00'00'00'00'00'00'00'00'00'00'	10CC			
00'00'00'00'00'00'00'00'00'00'00'00'	10D8			
00'00'00'00'00'00'00'00'00'00'00'00'	10E4			
00'00'00'00'00'00'00'00'00'00'00'00'	10F0			
00'00'00'00'00'00'00'00'00'00'00'00'	10FC			
00'00'00'00'00'00'00'00'00'00'00'00'	1108			
00'00'00'00'00'00'00'00'00'00'00'00'	1114			
00'00'00'00'00'00'00'00'00'00'00'00'	1120			
00'00'00'00'00'00'00'00'00'00'00'00'	112C			
00'00'00'00'00'00'00'00'00'00'00'00'	1138			
00'00'00'00'00'00'00'00'00'00'00'00'	1144			
00'00'00'00'00'00'00'00'00'00'00'00'	1150			
00'00'00'00'00'00'00'00'00'00'00'00'	115C			
00'00'00'00'00'00'00'00'00'00'00'00'	1168			
00'00'00'00'00'00'00'00'00'00'00'00'	1174			
00'00'00'00'00'00'00'00'00'00'00'00'	1180			

VMS\$VECTOR
4-003

- Define entry vectors for VMSRTL B 12

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 57
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (29)

00'00'00'00'00'00'00'00'00'00'00'00' 118C
00'00'00'00'00'00'00'00'00'00'00'00' 1198
00'00'00'00'00'00'00'00'00'00'00'00' 11A4
00'00'00'00'00'00'00'00'00'00'00'00' 11B0
00'00'00'00'00'00'00'00'00'00'00'00' 11BC
00'00'00'00'00'00'00'00'00'00'00'00' 11C8
00'00'00'00'00'00'00'00'00'00'00'00' 11D4
00'00'00'00'00'00'00'00'00'00'00'00' 11E0
00'00'00'00'00'00'00'00'00'00'00'00' 11EC
00'00'00'00'00'00'00'00'00'00'00'00' 11F8
00'00'00'00'00'00'00'00'00'00'00'00' 1204
00'00'00'00'00'00'00'00'00'00'00'00' 1210
00'00'00'00'00'00'00'00'00'00'00'00' 121C
00'00'00'00'00'00'00'00'00'00'00'00' 1228
00'00'00'00'00'00'00'00'00'00'00'00' 1234
00'00'00'00'00'00'00'00'00'00'00'00' 1240
00'00'00'00'00'00'00'00'00'00'00'00' 124C
00'00'00'00'00'00'00'00'00'00'00'00' 1258
00'00'00'00'00'00'00'00'00'00'00'00' 1264
00'00'00'00'00'00'00'00'00'00'00'00' 1270
00'00'00'00'00'00'00'00'00'00'00'00' 127C
00'00'00'00'00'00'00'00'00'00'00'00' 1288
00'00'00'00'00'00'00'00'00'00'00'00' 1294
00'00'00'00'00'00'00'00'00'00'00'00' 12A0
00'00'00'00'00'00'00'00'00'00'00'00' 12AC
00'00'00'00'00'00'00'00'00'00'00'00' 12B8
00'00'00'00'00'00'00'00'00'00'00'00' 12C4
00'00'00'00'00'00'00'00'00'00'00'00' 12D0
00'00'00'00'00'00'00'00'00'00'00'00' 12DC
00'00'00'00'00'00'00'00'00'00'00'00' 12E8
00'00'00'00'00'00'00'00'00'00'00'00' 12F4
00'00'00'00'00'00'00'00'00'00'00'00' 1300
00'00'00'00'00'00'00'00'00'00'00'00' 130C

1318 2539
1318 2540

MAC CALL COB COB\$HANDLER COB\$\$HANDLER

```
1320 2542 : MODULE COB$IOEXCEPTION -- I/O error processing
1320 2543       MAC      CALL      COB      COB$IOEXCEPTION
1328 2544
1328 2545 : MODULE COB$ERROR -- Process compiled-code-detected errors
1328 2546       MAC      CALL      COB      COB$ERROR
1330 2547
1330 2548 : MODULE COB$INTARI -- Intermediate Data Type Arithmetic
1330 2549       MAC      CALL      COB      COB$ADDI
1338 2550       MAC      CALL      COB      COB$SUBI
1340 2551       MAC      CALL      COB      COB$MULI
1348 2552       MAC      CALL      COB      COB$DIVI
1350 2553       MAC      CALL      COB      COB$DIVI_OSE
1358 2554       MAC      CALL      COB      COB$CMPI
1360 2555
1360 2556 : MODULE COB$INTER -- Conversions to and from Intermediate Data Type
1360 2557       MAC      JSB      COB      COB$CVTDI_R7
1368 2558       MAC      JSB      COB      COB$CVTFI_R7
1370 2559       MAC      JSB      COB      COB$CVTID_R7
1378 2560       MAC      JSB      COB      COB$CVTIF_R7
1380 2561       MAC      JSB      COB      COB$CVTIL_R8
1388 2562       MAC      JSB      COB      COB$CVTIP_R9
1390 2563       MAC      JSB      COB      COB$CVTIQ_R8
1398 2564       MAC      JSB      COB      COB$CVTIW_R8
13A0 2565       MAC      JSB      COB      COB$CVTLI_R8
13A8 2566       MAC      JSB      COB      COB$CVTPI_R9
13B0 2567       MAC      JSB      COB      COB$CVTQI_R8
13B8 2568       MAC      JSB      COB      COB$CVTRIC_R8
13C0 2569       MAC      JSB      COB      COB$CVTRIP_R9
13C8 2570       MAC      JSB      COB      COB$CVTRIQ_R8
13D0 2571       MAC      JSB      COB      COB$CVTRIW_R8
13D8 2572       MAC      JSB      COB      COB$CVTTI_R8
13E0 2573       MAC      JSB      COB      COB$CVTWI_R8
13E8 2574
13E8 2575 : MODULE COB$ACC_DATE -- Support for ACCEPT DATE
13E8 2576       MAC      CALL      COB      COB$ACC_DATE
13F0 2577
13F0 2578 : MODULE COB$ACC_DAY -- Support for ACCEPT DAY
13F0 2579       MAC      CALL      COB      COB$ACC_DAY
13F8 2580
13F8 2581 : MODULE COB$ACC_DAYWEEK -- ACCEPT DAY OF WEEK
13F8 2582       MAC      CALL      COB      COB$ACC_DAYWEEK
1400 2583
1400 2584 : MODULE COB$ACC_TIME -- Support for ACCEPT TIME
1400 2585       MAC      CALL      COB      COB$ACC_TIME
1408 2586
1408 2587 : MODULE COB$ACCEPT -- Support for ACCEPT
1408 2588       MAC      CALL      COB      COB$ACCEPT
1410 2589
1410 2590 : MODULE COB$DISPLAY -- Support for DISPLAY and DISPLAY WITH NO ADVANCING
1410 2591       MAC      CALL      COB      COB$DISPLAY
1418 2592       MAC      CALL      COB      COB$DISP_NO_ADV
1420 2593
1420 2594 : MODULE COB$DIVQ_R8 -- Quadword division
1420 2595       MAC      JSB      COB      COB$DIVQ_R8
1428 2596
1428 2597 : MODULE COB$MULQ_R8 -- Quadword multiplication
1428 2598       MAC      JSB      COB      COB$MULQ_R8
```

```
1430 2599
1430 2600 ; MODULE COB$PAUSE -- Support for STOP
1430 2601      MAC      CALL      COB      COB$PAUSE
1438 2602
1438 2603 ; MODULE COB$CVTPQ_R9 -- Packed to Quad conversion
1438 2604      MAC      JSB      COB      COB$CVTPQ_R9
1440 2605
1440 2606 ; MODULE COB$CVTQP_R9 -- Quad to Packed conversion
1440 2607      MAC      JSB      COB      COB$CVTQP_R9
1448 2608
1448 2609 ; MODULE COB$CVTRPQ_R9 -- Rounded Packed to Quad conversion
1448 2610      MAC      JSB      COB      COB$CVTRPQ_R9
1450 2611
1450 2612 ; MODULE COB$CVTRQP_R9 -- Rounded Quad to Packed conversion
1450 2613      MAC      JSB      COB      COB$CVTRQP_R9
1458 2614
1458 2615
1458 2616
```

```
1458 2618 ; MODULE COB$MSGDEF -- Defines COB$ all conditon codes
1458 2619 MAC SYM COB COB$-CALFAI
1458 2620 MAC SYM COB COB$-CANFAIL
1458 2621 MAC SYM COB COB$-DELINCOPE
1458 2622 MAC SYM COB COB$-DELNO R S
1458 2623 MAC SYM COB COB$-DELUNDFIL
1458 2624 MAC SYM COB COB$-DISMORMAX
1458 2625 MAC SYM COB COB$-EOFON ACC
1458 2626 MAC SYM COB COB$-ERRDURACC
1458 2627 MAC SYM COB COB$-ERRDURDIS
1458 2628 MAC SYM COB COB$-ERRDURSOR
1458 2629 MAC SYM COB COB$-ERRON FIL
1458 2630 MAC SYM COB COB$-EXPDBOVER
1458 2631 MAC SYM COB COB$-FAIFREEVM
1458 2632 MAC SYM COB COB$-FAIGET-EF
1458 2633 MAC SYM COB COB$-FAIGET-VM
1458 2634 MAC SYM COB COB$-FATINTERR
1458 2635 MAC SYM COB COB$-FILALRCLO
1458 2636 MAC SYM COB COB$-FILALRLOC
1458 2637 MAC SYM COB COB$-FILALROPE
1458 2638 MAC SYM COB COB$-FILCLOLOC
1458 2639 MAC SYM COB COB$-FILNOTFOU
1458 2640 MAC SYM COB COB$-GOTO ALT
1458 2641 MAC SYM COB COB$-INTDIVZER
1458 2642 MAC SYM COB COB$-INTEXPOVE
1458 2643 MAC SYM COB COB$-INTEXPUND
1458 2644 MAC SYM COB COB$-INTRESOPE
1458 2645 MAC SYM COB COB$-INVARG
1458 2646 MAC SYM COB COB$-INVCHANAM
1458 2647 MAC SYM COB COB$-INVDECDIG
1458 2648 MAC SYM COB COB$-INVLINVAL
1458 2649 MAC SYM COB COB$-KEYNOTMAT
1458 2650 MAC SYM COB COB$-LSTHNDLDB
1458 2651 MAC SYM COB COB$-LSTHNDUSE
1458 2652 MAC SYM COB COB$-NAMNOTLIN
1458 2653 MAC SYM COB COB$-NESERRPER
1458 2654 MAC SYM COB COB$-NORMAL
1458 2655 MAC SYM COB COB$-NO_NEXLOG
1458 2656 MAC SYM COB COB$-NO_NEXVAL
1458 2657 MAC SYM COB COB$-NO_SPACE
1458 2658 MAC SYM COB COB$-NO_USEPRO
1458 2659 MAC SYM COB COB$-OCCDEPOVE
1458 2660 MAC SYM COB COB$-OPTMISCLO
1458 2661 MAC SYM COB COB$-OPTMISOPE
1458 2662 MAC SYM COB COB$-OPTMISREA
1458 2663 MAC SYM COB COB$-OPTMISSTA
1458 2664 MAC SYM COB COB$-ORGNOTMAT
1458 2665 MAC SYM COB COB$-PRIKEYCHA
1458 2666 MAC SYM COB COB$-REAINCOPE
1458 2667 MAC SYM COB COB$-REASMAMIN
1458 2668 MAC SYM COB COB$-REAUNOFIL
1458 2669 MAC SYM COB COB$-REACTPER
1458 2670 MAC SYM COB COB$-REACTUSE
1458 2671 MAC SYM COB COB$-RECLOCDEL
1458 2672 MAC SYM COB COB$-RECLOCREA
1458 2673 MAC SYM COB COB$-RECLOCREW
1458 2674 MAC SYM COB COB$-RECLOCSTA
```

- Define entry vectors for VMSRTL F 12

16-SEP-1984 02:15:59
6-SEP-1984 11:48:04

VAX/VMS Macro V04-00
[VMSRTL.SRC]VMSVECTOR.MAR:1

Page 61
(31)

1458	2675	MAC	SYM	COB	COBS-RECLOCWRI
1458	2676	MAC	SYM	COB	COBS-RECLOC OK
1458	2677	MAC	SYM	COB	COBS-RECNOTEXI
1458	2678	MAC	SYM	COB	COBS-RECNOTLOC
1458	2679	MAC	SYM	COB	COBS-REWCREDUP
1458	2680	MAC	SYM	COB	COBS-REWDISDUP
1458	2681	MAC	SYM	COB	COBS-REWINCOPE
1458	2682	MAC	SYM	COB	COBS-REWNO R S
1458	2683	MAC	SYM	COB	COBS-REWSMAMIN
1458	2684	MAC	SYM	COB	COBS-REWUNOFIL
1458	2685	MAC	SYM	COB	COBS-SETEXTFAI
1458	2686	MAC	SYM	COB	COBS-STAINCOPE
1458	2687	MAC	SYM	COB	COBS-STAUONFIL
1458	2688	MAC	SYM	COB	COBS-SUBOVOLON
1458	2689	MAC	SYM	COB	COBS-TIMOVOLON
1458	2690	MAC	SYM	COB	COBS-UNDEF EXP
1458	2691	MAC	SYM	COB	COBS-UNEINSCON
1458	2692	MAC	SYM	COB	COBS-UNLNO CUR
1458	2693	MAC	SYM	COB	COBS-UNLUNOFIL
1458	2694	MAC	SYM	COB	COBS-WRIBEYBOU
1458	2695	MAC	SYM	COB	COBS-WRICREDUP
1458	2696	MAC	SYM	COB	COBS-WRIDISDUP
1458	2697	MAC	SYM	COB	COBS-WRIDUPALT
1458	2698	MAC	SYM	COB	COBS-WRIDUPKEY
1458	2699	MAC	SYM	COB	COBS-WRIINCOPE
1458	2700	MAC	SYM	COB	COBS-WRINOTASC
1458	2701	MAC	SYM	COB	COBS-WRISMAMIN
1458	2702	MAC	SYM	COB	COBS-WRIUNOFIL

[illegible]

```
1458 2704 :+
1458 2705 : The following routines are N O T in the sharable library. They
1458 2706 : are tabulated here to provide a complete tabulation of all entry
1458 2707 : points known to COBOL when this module is assembled in the "ALLGBL"
1458 2708 : mode.
1458 2709 :-
1458 2710 :
1458 2711 : MODULE COB$CALL -- Support COBOL CALL
1458 2712 :     MAC      NOVECT COB      COB$CALL
1458 2713 :
1458 2714 : MODULE COB$CANCEL -- Support COBOL CANCEL
1458 2715 :     MAC      NOVECT COB      COB$CANCEL
1458 2716 :
1458 2717 : MODULE COB$CNVOUT -- Support conversion routines
1458 2718 :     MAC      NOVECT COB      COB$CNVOUT
1458 2719 :
1458 2720 : MODULE COB$CVTDP -- Convert Double to Packed
1458 2721 :     MAC      NOVECT COB      COB$CVTDP_R9
1458 2722 :
1458 2723 : MODULE COB$CVTRDP -- Convert Rounded Double to Packed
1458 2724 :     MAC      NOVECT COB      COB$CVTRDP_R9
1458 2725 :
1458 2726 : MODULE COB$CVTPD -- Convert Packed to Double
1458 2727 :     MAC      NOVECT COB      COB$CVTPD_R9
1458 2728 :
1458 2729 : MODULE COB$CVTDQ -- Convert Double to Quadword
1458 2730 :     MAC      NOVECT COB      COB$CVTDQ_R8
1458 2731 :
1458 2732 : MODULE COB$CVTRDQ -- Convert Rounded Double to Quadword
1458 2733 :     MAC      NOVECT COB      COB$CVTRDQ_R8
1458 2734 :
1458 2735 : MODULE COB$CVTQD -- Convert Quadword to Double
1458 2736 :     MAC      NOVECT COB      COB$CVTQD_R8
1458 2737 :
1458 2738 : MODULE COB$CVTFP -- Convert Floating to Packed
1458 2739 :     MAC      NOVECT COB      COB$CVTFP_R9
1458 2740 :
1458 2741 : MODULE COB$CVTRFP -- Convert Rounded Floating to Packed
1458 2742 :     MAC      NOVECT COB      COB$CVTRFP_R9
1458 2743 :
1458 2744 : MODULE COB$CVTPF -- Convert Packed to Floaing
1458 2745 :     MAC      NOVECT COB      COB$CVTPF_R9
1458 2746 :
1458 2747 : MODULE COB$CVTFQ -- Convert Floating to Quadword
1458 2748 :     MAC      NOVECT COB      COB$CVTFQ_R8
1458 2749 :
1458 2750 : MODULE COB$CVTRFQ -- Convert Rounded Floating to Quadword
1458 2751 :     MAC      NOVECT COB      COB$CVTRFQ_R8
1458 2752 :
1458 2753 : MODULE COB$CVTQF -- Convert Quadword to Floating
1458 2754 :     MAC      NOVECT COB      COB$CVTQF_R8
1458 2755 :
1458 2756 : MODULE COB$EXPI -- CIT exponentiation
1458 2757 :     MAC      NOVECT COB      COB$EXPI
1458 2758 :     MAC      NOVECT COB      COB$EXPI_OSE
1458 2759 :
1458 2760 : MODULE COB$LINAGE -- Support LINAGE
```

```

1458 2761      MAC      NOVECT  COB      COB$LINAGE
1458 2762      MAC      NOVECT  COB      COB$INIT_LINAGE
1458 2763      MAC      NOVECT  COB      COB$TERM_LINAGE
1458 2764
1458 2765 : MODULE COB$RMS_BLOCKS -- RMS data block
1458 2766      MAC      NOVECT  COB      COB$AB_NAM
1458 2767
1458 2768 : MODULE COB$SET_SWITCH -- Set external switches
1458 2769      MAC      NOVECT  COB      COB$SET_SWITCH
1458 2770
1458 2771 : MODULE COB$SWITCH -- Support switches
1458 2772      MAC      NOVECT  COB      COB$SWITCH
1458 2773
1458 2774 : MODULE LIB$AB_ASC_EBC -- ASCII to EBCDIC translation table
1458 2775      MAC      NOVECT  LIB      LIB$AB_ASC_EBC
1458 2776
1458 2777 : MODULE LIB$AB_CVT_O_U -- Overpunch to Unsigned translation table
1458 2778      MAC      NOVECT  LIB      LIB$AB_CVT_O_U
1458 2779
1458 2780 : MODULE LIB$AB_EBC_ASC -- EBCDIC to ASCII translation table
1458 2781      MAC      NOVECT  LIB      LIB$AB_EBC_ASC
1458 2782

```

```
1458 2784 : POST VMS VERSION 2.0 ADDITIONS
1458 2785 : -----
1458 2786 :
1458 2787 : This point marks where the modules added after VMS Version 2.0 are
1458 2788 : placed (unless they fit in some existing hole).
1458 2789 : The vector entries to the shared components come first, and
1458 2790 : are then followed by the non-vectorized (non-shared) entries and the new
1458 2791 : symbol definitions.
1458 2792 : There are two flavors of shared components -- those that are
1458 2793 : vectored and those that aren't. Those that aren't are in the shared
1458 2794 : image solely for the purpose of binding of VMSRTL.EXE, but are not
1458 2795 : accessible from outside of the image.
1458 2796 :
1458 2797 : SHARED COMPONENTS ( VECTORED )
1458 2798 : -----
1458 2799 :
1458 2800 :
1458 2801 : MODULE LIB$LUN
1458 2802 :     MAC      CALL      LIB      LIB$FREE_LUN
1460 2803 :     MAC      CALL      LIB      LIB$GET_LUN
1468 2804 :
1468 2805 : MODULE LIB$EF
1468 2806 :     MAC      CALL      LIB      LIB$FREE_EF
1470 2807 :     MAC      CALL      LIB      LIB$GET_EF
1478 2808 :     MAC      CALL      LIB      LIB$RESERVE_EF
1480 2809 :
1480 2810 : MODULE LIB$ANALYZE_SDESC
1480 2811 :     MAC      CALL      LIB      LIB$ANALYZE_SDESC
1488 2812 :     MAC      JSB       LIB      LIB$ANALYZE_SDESC_R2
1490 2813 :
1490 2814 : MODULE STR$ANALYZE_SDESC
1490 2815 :     MAC      CALL      STR      STR$ANALYZE_SDESC
1498 2816 :     MAC      JSB       STR      STR$ANALYZE_SDESC_R1
14A0 2817 :
14A0 2818 : MODULE LIB$FILESCAN -- find files matching wild-card description
14A0 2819 :     MAC      CALL      LIB      LIB$FILE_SCAN
14A8 2820 :     MAC      CALL      LIB      LIB$FIND_FILE
14B0 2821 :
14B0 2822 :
14B0 2823 :
14B0 2824 : SHARED COMPONENTS ( NON - VECTORED )
14B0 2825 : -----
14B0 2826 :
14B0 2827 : MODULE STR$$CHECK STATUS
14B0 2828 :     MAC      NOVECT  STR      STR$$CHECK_STATUS_R2
14B0 2829 :
14B0 2830 :
14B0 2831 : NON - SHARED COMPONENTS
14B0 2832 : -----
14B0 2833 :
14B0 2834 : MODULE LIB$AB_ASC_EBC_REV -- Reversible ASCII to EBCDIC trans. table
14B0 2835 :     MAC      NOVECT  LIB      LIB$AB_ASC_EBC_REV
14B0 2836 :
14B0 2837 : MODULE LIB$AB_EBC_ASC_REV -- Reversible EBCDIC to ASCII trans. table
14B0 2838 :     MAC      NOVECT  LIB      LIB$AB_EBC_ASC_REV
14B0 2839 :
14B0 2840 : MODULE LIB$AB_CVTPT_Z -- packed decimal to zoned translation table
```

```
1480 2841      MAC      NOVECT LIB      LIB$AB_CVTPT_Z
1480 2842
1480 2843 ; MODULE LIB$AB_CVTTP_Z -- zoned to packed decimal translation table
1480 2844      MAC      NOVECT LIB      LIB$AB_CVTTP_Z
1480 2845
1480 2846 ; MODULE LIB$CALLG -- execute CALLG instruction
1480 2847      MAC      NOVECT LIB      LIB$CALLG
1480 2848
1480 2849 ; MODULE LIB$DECODE_FAULT -- decode instruction stream
1480 2850      MAC      NOVECT LIB      LIB$DECODE_FAULT
1480 2851
1480 2852 ; MODULE LIB$EDIV -- execute EDIV instruction
1480 2853      MAC      NOVECT LIB      LIB$EDIV
1480 2854
1480 2855 ; MODULE LIB$EMUL -- execute EMUL instruction
1480 2856      MAC      NOVECT LIB      LIB$EMUL
1480 2857
1480 2858 ; MODULE LIB$MOVC3 -- execute MOVC3 instruction
1480 2859      MAC      NOVECT LIB      LIB$MOVC3
1480 2860
1480 2861 ; MODULE LIB$MOVC5 -- execute MOVC5 instruction
1480 2862      MAC      NOVECT LIB      LIB$MOVC5
1480 2863
1480 2864 ; MODULE COB$AB_DEEDIT -- COBOL translation table for 'de-editing
1480 2865      MAC      NOVECT COB      COB$AB_DEEDIT
1480 2866
1480 2867 ; MODULE COB$DBEXCEPTION -- COBOL Data Base Exception Processing
1480 2868      MAC      NOVECT COB      COB$DBEXCEPTION
1480 2869
1480 2870 ; MODULE COB$AB_SPANC2 -- COBOL SPANC translation table II
1480 2871      MAC      NOVECT COB      COB$AB_SPANC2
1480 2872
1480 2873 ; MODULE LIB$CVTDXDX -- LIB general data type conversion routine
1480 2874      MAC      NOVECT LIB      LIB$CVT_DX_DX
1480 2875
1480 2876 ; MODULE LIB$$PACK_ARITH -- perform packed arithmetic for STR$DIVIDE
1480 2877      MAC      NOVECT LIB      LIB$$CVT_STR_PACK_R9
1480 2878      MAC      NOVECT LIB      LIB$$CALC_D_R7
1480 2879      MAC      NOVECT LIB      LIB$$CALC_Q_R9
1480 2880      MAC      NOVECT LIB      LIB$$ADJUST_Q_R9
1480 2881      MAC      NOVECT LIB      LIB$$MUL_PACK_R10
1480 2882      MAC      NOVECT LIB      LIB$$SUB_PACK_R8
1480 2883      MAC      NOVECT LIB      LIB$$ROUND_R7
1480 2884      MAC      NOVECT LIB      LIB$$CVT_PACK_STR_R8
1480 2885
1480 2886 ; MODULE STR$ARITH -- string arithmetic (added entry point STR$DIVIDE)
1480 2887      MAC      NOVECT STR      STR$DIVIDE
1480 2888
1480 2889 ; MODULE FOR$INIUND -- FORTRAN underflow handler initialization
1480 2890      MAC      NOVECT FOR      FOR$INIT_UNDER
1480 2891
1480 2892 ; MODULE FOR$UNDERF -- FORTRAN underflow handler
1480 2893      MAC      NOVECT FOR      FOR$UNDERFLOW_HANDLER
1480 2894
1480 2895 ; MODULE OTS$POWLULU -- unsigned ** unsigned integer power
1480 2896      MAC      NOVECT OTS      OTS$POWLULU
1480 2897
```

```
1480 2898 : MODULE STR$COMPARE_CASE_BLIND -- Compare strings case-blind
1480 2899 :         MAC      NOVECT STR      STR$CASE_BLIND_COMPARE
1480 2900 :
1480 2901 : MODULE STR$FIND_FIRST -- Find 1st char in or not in set
1480 2902 :         MAC      NOVECT STR      STR$FIND_FIRST_IN_SET
1480 2903 :         MAC      NOVECT STR      STR$FIND_FIRST_NOT_IN_SET
1480 2904 :
1480 2905 : MODULE STR$FIND_FIRST_SUBSTRING -- Find first substring
1480 2906 :         MAC      NOVECT STR      STR$FIND_FIRST_SUBSTRING
1480 2907 :
1480 2908 : NEW ENTRY POINTS FOR VAX BASIC 2.0
1480 2909 : -----
1480 2910 :
1480 2911 : MODULE BAS$CVTTP -- Basic convert text to packed
1480 2912 :         MAC      CALL      BAS      BAS$CVT_T_P
1488 2913 :
1488 2914 : MODULE LIB$$ADDP -- add packed instruction for BAS$CVT_T_P
1488 2915 :         MAC      NOVECT LIB      LIB$$ADDP_R7
1488 2916 :
1488 2917 : MODULE BAS$UPI_TERM_IO
1488 2918 : (use up addr vacated by LIB$$ADDP_R7)
1488 2919 :         MAC      CALL      BAS      BAS$IN_B_R
14C0 2920 :
14C0 2921 : MODULE OT$$CNVOUT -- convert floating to E formatted text
14C0 2922 : (shared, not vectored)
14C0 2923 :         MAC      NOVECT OTS      OT$$CNVOUT
14C0 2924 :
14C0 2925 : OLD ENTRY POINTS FOR MODULE BAS$CVTOUT, originally overlooked
14C0 2926 :         MAC      CALL      BAS      BAS$CVT_OUT_F_E
14C8 2927 :         MAC      CALL      BAS      BAS$CVT_OUT_F_F
14D0 2928 :         MAC      CALL      BAS      BAS$CVT_OUT_D_E
14D8 2929 :         MAC      CALL      BAS      BAS$CVT_OUT_D_F
14E0 2930 :         MAC      CALL      BAS      BAS$CVT_OUT_D_G
14E8 2931 :
14E8 2932 : NEW ENTRY POINTS FOR MODULE BAS$CVTOUT, Basic output conversion
14E8 2933 :         MAC      CALL      BAS      BAS$CVT_OUT_G_E
14F0 2934 :         MAC      CALL      BAS      BAS$CVT_OUT_G_F
14F8 2935 :         MAC      CALL      BAS      BAS$CVT_OUT_G_G
1500 2936 :         MAC      CALL      BAS      BAS$CVT_OUT_H_E
1508 2937 :         MAC      CALL      BAS      BAS$CVT_OUT_H_F
1510 2938 :         MAC      CALL      BAS      BAS$CVT_OUT_H_G
1518 2939 :         MAC      CALL      BAS      BAS$CVT_OUT_P_E
1520 2940 :         MAC      CALL      BAS      BAS$CVT_OUT_P_F
1528 2941 :         MAC      CALL      BAS      BAS$CVT_OUT_P_G
1530 2942 :
1530 2943 : NEW ENTRY POINTS FOR MODULE BAS$CMPAPP, Basic compare approximate
1530 2944 :         MAC      CALL      BAS      BAS$CMPG_APP
1538 2945 :         MAC      CALL      BAS      BAS$CMPH_APP
1540 2946 :
1540 2947 : NEW ENTRY POINTS FOR MODULE BAS$COPYFD, Basic copy floating
1540 2948 : (shared, not vectored)
1540 2949 :         MAC      NOVECT BAS      BAS$COPY_G_R1
1540 2950 :         MAC      NOVECT BAS      BAS$COPY_H_R3
1540 2951 :
1540 2952 : NEW ENTRY POINTS FOR MODULE BAS$NUM, Basic NUM function
1540 2953 :         MAC      CALL      BAS      BAS$NUM_G
1548 2954 :         MAC      CALL      BAS      BAS$NUM_H
```

```
1550 2955      MAC      CALL      BAS      BAS$NUM_P
1558 2956
1558 2957 ; NEW ENTRY POINTS FOR MODULE BAS$NUM1, Basic NUM1 function
1558 2958      MAC      CALL      BAS      BAS$NUM1_G
1560 2959      MAC      CALL      BAS      BAS$NUM1_H
1568 2960      MAC      CALL      BAS      BAS$NUM1_P
1570 2961
1570 2962 ; NEW ENTRY POINTS FOR MODULE BAS$STR, Basic STR$ function
1570 2963      MAC      CALL      BAS      BAS$STR_G
1578 2964      MAC      CALL      BAS      BAS$STR_H
1580 2965      MAC      CALL      BAS      BAS$STR_P
1588 2966
1588 2967 ; NEW ENTRY POINTS FOR MODULE BAS$UPI_TERM_IO, Basic UPI level I/O
1588 2968      MAC      CALL      BAS      BAS$OUT_G_V_S
1590 2969      MAC      CALL      BAS      BAS$OUT_G_V_B
1598 2970      MAC      CALL      BAS      BAS$OUT_G_V_C
15A0 2971      MAC      CALL      BAS      BAS$OUT_H_V_S
15A8 2972      MAC      CALL      BAS      BAS$OUT_H_V_B
15B0 2973      MAC      CALL      BAS      BAS$OUT_H_V_C
15B8 2974      MAC      CALL      BAS      BAS$OUT_P_DX_S
15C0 2975      MAC      CALL      BAS      BAS$OUT_P_DX_B
15C8 2976      MAC      CALL      BAS      BAS$OUT_P_DX_C
15D0 2977      MAC      CALL      BAS      BAS$IN_G_R
15D8 2978      MAC      CALL      BAS      BAS$IN_H_R
15E0 2979      MAC      CALL      BAS      BAS$IN_P_DX
15E8 2980
15E8 2981 ; NEW ENTRY POINTS FOR BAS$VAL, Basic VAL function
15E8 2982      MAC      CALL      BAS      BAS$VAL_G
15F0 2983      MAC      CALL      BAS      BAS$VAL_H
15F8 2984      MAC      CALL      BAS      BAS$VAL_P
1600 2985
1600 2986 ; MODULE BAS$$REC_PROC
1600 2987 ;   this is needed for BAS$ANSI_TAB, a non-shared entry point
1600 2988      MAC      JSB      BAS      BAS$$REC_WSL1
1608 2989
1608 2990 ; MODULE BAS$FIND, new entry point
1608 2991      MAC      CALL      BAS      BAS$FIND_RFA
1610 2992
1610 2993 ; MODULE BAS$GET, new entry point
1610 2994      MAC      CALL      BAS      BAS$GET_RFA
1618 2995
1618 2996 ; MODULE BAS$GETRFA, new
1618 2997      MAC      CALL      BAS      BAS$GETRFA
1620 2998
1620 2999 ; MODULE BAS$CB, old entry point must be vectored for improved BASKILL
1620 3000      MAC      CALL      BAS      BAS$$NEXT_LUN
1628 3001
1628 3002 ; MODULE BAS$IOBEG, new entry point
1628 3003      MAC      CALL      BAS      BAS$ANSI_INPUT
1630 3004
1630 3005 ; MODULE BAS$IOEND, new entry point
1630 3006      MAC      CALL      BAS      BAS$ANSI_IO_END
1638 3007
1638 3008 ; MODULE BAS$CTRLC, all entry points
1638 3009      MAC      CALL      BAS      BAS$CTRLC
1640 3010      MAC      CALL      BAS      BAS$RCTRLC
1648 3011      MAC      CALL      BAS      BAS$$CTRLC_INIT
```

VMSSVECTOR
4-003

- Define entry vectors for VMSRTL M 12

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 68
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (33)

1650 3012

```

1650 3014 .SBTTL MTH$$AB ALOG - Table for ALOG routines
1650 3015
1650 3016 :
1650 3017 : The MTH$$AB ALOG table is accessed by the low order exponent bit and the
1650 3018 : first 7 fraction bits (not including the hidden bit) of the argument. The
1650 3019 : table entries are offsets into the F FHI table. Note that the
1650 3020 : MTH$$AB ALOG table is data type independent and is used by all four LOG
1650 3021 : routines.
1650 3022 :
1650 3023 : This table is a duplicate of that in MTHALOG.MAR, but must remain
1650 3024 : separate.
1650 3025
1650 3026
1650 3027 MTH$$AB ALOG:
04 04 04 04 04 04 04 04 1650 3028 .BYTE ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04
04 04 04 04 04 04 04 04 1658 3029 .BYTE ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04, ^X04
09 09 09 04 04 04 04 04 1660 3030 .BYTE ^X04, ^X04, ^X04, ^X04, ^X04, ^X09, ^X09, ^X09
09 09 09 09 09 09 09 09 1668 3031 .BYTE ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09
09 09 09 09 09 09 09 09 1670 3032 .BYTE ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09, ^X09
0E 0E 0E 0E 0E 0E 0E 0E 1678 3033 .BYTE ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E
0E 0E 0E 0E 0E 0E 0E 0E 1680 3034 .BYTE ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E, ^X0E
13 13 13 13 13 13 13 13 1688 3035 .BYTE ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X13
18 18 13 13 13 13 13 13 1690 3036 .BYTE ^X13, ^X13, ^X13, ^X13, ^X13, ^X13, ^X18, ^X18
18 18 18 18 18 18 18 18 1698 3037 .BYTE ^X18, ^X18, ^X18, ^X18, ^X18, ^X18, ^X18, ^X18
1D 1D 1D 1D 1D 1D 1D 18 16A0 3038 .BYTE ^X18, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D, ^X1D
22 22 22 22 22 22 1D 1D 16A8 3039 .BYTE ^X1D, ^X1D, ^X22, ^X22, ^X22, ^X22, ^X22, ^X22
2C 27 27 27 27 27 27 22 16B0 3040 .BYTE ^X22, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27, ^X2C
31 31 31 31 2C 2C 2C 2C 16B8 3041 .BYTE ^X2C, ^X2C, ^X2C, ^X2C, ^X31, ^X31, ^X31, ^X31
45 40 40 3B 3B 36 36 36 16C0 3042 .BYTE ^X36, ^X36, ^X36, ^X3B, ^X3B, ^X40, ^X40, ^X45
FF FF FF FF FF FF FF FF 16C8 3043 .BYTE ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF, ^XFF
32 37 3C 41 FF FF FF FF 16D0 3044 .BYTE ^XFF, ^XFF, ^XFF, ^XFF, ^X41, ^X3C, ^X37, ^X32
23 23 28 28 28 2D 2D 32 16D8 3045 .BYTE ^X32, ^X2D, ^X2D, ^X28, ^X28, ^X28, ^X23, ^X23
19 1E 1E 1E 1E 1E 23 23 16E0 3046 .BYTE ^X23, ^X23, ^X1E, ^X1E, ^X1E, ^X1E, ^X1E, ^X19
14 14 19 19 19 19 19 19 16E8 3047 .BYTE ^X19, ^X19, ^X19, ^X19, ^X19, ^X19, ^X14, ^X14
OF 14 14 14 14 14 14 14 16F0 3048 .BYTE ^X14, ^X14, ^X14, ^X14, ^X14, ^X14, ^X14, ^X0F
OF OF OF OF OF OF OF OF 16F8 3049 .BYTE ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F, ^X0F
0A 0A 0A 0A 0A 0F 0F 0F 1700 3050 .BYTE ^X0F, ^X0F, ^X0F, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A
0A 0A 0A 0A 0A 0A 0A 0A 1708 3051 .BYTE ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A, ^X0A
05 05 05 05 0A 0A 0A 0A 1710 3052 .BYTE ^X0A, ^X0A, ^X0A, ^X0A, ^X05, ^X05, ^X05, ^X05
05 05 05 05 05 05 05 05 1718 3053 .BYTE ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05
05 05 05 05 05 05 05 05 1720 3054 .BYTE ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05, ^X05
00 00 00 00 05 05 05 05 1728 3055 .BYTE ^X05, ^X05, ^X05, ^X05, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1730 3056 .BYTE ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1738 3057 .BYTE ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1740 3058 .BYTE ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00
00 00 00 00 00 00 00 00 1748 3059 .BYTE ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00, ^X00

```

```
.SBTTL MTH$$AB_ATAN - Table for ATAN routines

1750 3061
1750 3062
1750 3063 :
1750 3064 : The MTH$$AB_ATAN table is a table of byte entries used to obtain an index
1750 3065 : into the ATAN_TABLE. MTH$$AB_ATAN is indexed using the low order bits of
1750 3066 : the exponent field and the high order bits of the fraction field. The
1750 3067 : MTH$$AB_ATAN table is independent of the data type and is used by all of
1750 3068 : the arc tangent routines.
1750 3069 :
1750 3070 : This table is a duplicate of that in MTHATAN.MAR, but must remain
1750 3071 : separate.
1750 3072
1750 3073 MTH$$AB_ATAN:
1750 3074 .BYTE ^X00, ^X00, ^X00, ^X03, ^X03, ^X06, ^X06, ^X09
1758 3075 .BYTE ^X09, ^X09, ^X0C, ^X0C, ^X0C, ^X0F, ^X0F, ^X12
1760 3076 .BYTE ^X12, ^X12, ^X15, ^X15, ^X15, ^X18, ^X18, ^X18
1768 3077 .BYTE ^X1B, ^X1B, ^X1B, ^X1B, ^X1E, ^X1E, ^X1E, ^X21
1770 3078 .BYTE ^X21, ^X21, ^X21, ^X21, ^X24, ^X24, ^X24, ^X24
1778 3079 .BYTE ^X24, ^X24, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27
1780 3080 .BYTE ^X27, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27, ^X27
1787 3081
1787 3082 .END
```

VMSS\$VECTOR
Symbol table

- Define entry vectors for VMSRTL C 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 71
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMS\$VECTOR.MAR;1 (35)

\$\$BASS\$BLNK_LINE	00000BA0	RG	01	\$\$BASS\$ERN	00000988	RG	01
\$\$BASS\$CB_GET	00000B80	RG	01	\$\$BASS\$ERR	00000978	RG	01
\$\$BASS\$CB_POP	00000B70	RG	01	\$\$BASS\$ERROR	00000B50	RG	01
\$\$BASS\$CB_PUSH	00000B78	RG	01	\$\$BASS\$ERT	00000990	RG	01
\$\$BASS\$CLOSE_ALL	00000BF8	RG	01	\$\$BASS\$FIND	00000AC8	RG	01
\$\$BASS\$CTRLC_INIT	00001648	RG	01	\$\$BASS\$FIND_KEY	00000AD8	RG	01
\$\$BASS\$ERR_INIT	00000B88	RG	01	\$\$BASS\$FIND_RECORD	00000AD0	RG	01
\$\$BASS\$FORMAT_INT	00000BF0	RG	01	\$\$BASS\$FIND_RFA	00001608	RG	01
\$\$BASS\$NEXT_LON	00001620	RG	01	\$\$BASS\$FREE	00000B18	RG	01
\$\$BASS\$OPEN_ZERO	00000B90	RG	01	\$\$BASS\$GET	00000A90	RG	01
\$\$BASS\$RECOO_INIT	00000B98	RG	01	\$\$BASS\$GETRFA	00001618	RG	01
\$\$BASS\$REC_WSL1	00001600	RG	01	\$\$BASS\$GET_KEY	00000AA0	RG	01
\$\$BASS\$SCALE_L_R1	00000BD8	RG	01	\$\$BASS\$GET_RECORD	00000A98	RG	01
\$\$BASS\$SCALE_RT	00000BE0	RG	01	\$\$BASS\$GET_RFA	00001610	RG	01
\$\$BASS\$SIGNAL	00000BA8	RG	01	\$\$BASS\$HANDLER	00000998	RG	01
\$\$BASS\$SIGNAL_IO	00000BB0	RG	01	\$\$BASS\$INIT_DEF_R8	00000920	RG	01
\$\$BASS\$STATU_INIT	00000BB8	RG	01	\$\$BASS\$INIT_DFS_R8	00000928	RG	01
\$\$BASS\$STOP	00000BC0	RG	01	\$\$BASS\$INIT_GOSUB	00000930	RG	01
\$\$BASS\$STOP_IO	00000BC8	RG	01	\$\$BASS\$INIT_R8	00000918	RG	01
\$\$BASS\$STOP_RMS	00000BE8	RG	01	\$\$BASS\$INPUT	000009A0	RG	01
\$\$BASS\$UDF_RL1	00000C00	RG	01	\$\$BASS\$INPUT_LINE	000009B0	RG	01
\$\$BASS\$UDF_WL1	00000C08	RG	01	\$\$BASS\$INSTR	000008B0	RG	01
\$\$BASS\$ANSI_INPUT	00001628	RG	01	\$\$BASS\$IN_B_R	000014B8	RG	01
\$\$BASS\$ANSI_IO_END	00001630	RG	01	\$\$BASS\$IN_D_R	000009F0	RG	01
\$\$BASS\$BUFSTZ	00000B40	RG	01	\$\$BASS\$IN_F_R	000009E8	RG	01
\$\$BASS\$CANTYFAHEAD	00000BD0	RG	01	\$\$BASS\$IN_G_R	000015D0	RG	01
\$\$BASS\$CCPOS	00000B20	RG	01	\$\$BASS\$IN_H_R	000015D8	RG	01
\$\$BASS\$CHR	00000B48	RG	01	\$\$BASS\$IN_L_R	000009E0	RG	01
\$\$BASS\$CLOSE	00000A88	RG	01	\$\$BASS\$IN_P_DX	000015E0	RG	01
\$\$BASS\$CMPD_APP	00000890	RG	01	\$\$BASS\$IN_T_DX	000009F8	RG	01
\$\$BASS\$CMPF_APP	00000888	RG	01	\$\$BASS\$IN_W_R	000009D8	RG	01
\$\$BASS\$CMPI_APP	00001530	RG	01	\$\$BASS\$IO_END	000009D0	RG	01
\$\$BASS\$CMPI_APP	00001538	RG	01	\$\$BASS\$INPUT	000009A8	RG	01
\$\$BASS\$CTRLC	00001638	RG	01	\$\$BASS\$MAT_INPUT	00000A68	RG	01
\$\$BASS\$CVT_OUT_D_E	000014D0	RG	01	\$\$BASS\$MAT_LINPUT	00000A70	RG	01
\$\$BASS\$CVT_OUT_D_F	000014D8	RG	01	\$\$BASS\$MAT_PRINT	00000A60	RG	01
\$\$BASS\$CVT_OUT_D_G	000014E0	RG	01	\$\$BASS\$MAT_READ	00000A78	RG	01
\$\$BASS\$CVT_OUT_F_E	000014C0	RG	01	\$\$BASS\$NUMT_D	000008F0	RG	01
\$\$BASS\$CVT_OUT_F_F	000014C8	RG	01	\$\$BASS\$NUM1_F	000008E8	RG	01
\$\$BASS\$CVT_OUT_G_E	000014E8	RG	01	\$\$BASS\$NUM1_G	00001558	RG	01
\$\$BASS\$CVT_OUT_G_F	000014F0	RG	01	\$\$BASS\$NUM1_H	00001560	RG	01
\$\$BASS\$CVT_OUT_G_G	000014F8	RG	01	\$\$BASS\$NUM1_L	000008F8	RG	01
\$\$BASS\$CVT_OUT_H_E	00001500	RG	01	\$\$BASS\$NUM1_P	00001568	RG	01
\$\$BASS\$CVT_OUT_H_F	00001508	RG	01	\$\$BASS\$NUM_D	000008D8	RG	01
\$\$BASS\$CVT_OUT_H_G	00001510	RG	01	\$\$BASS\$NUM_F	000008D0	RG	01
\$\$BASS\$CVT_OUT_P_E	00001518	RG	01	\$\$BASS\$NUM_G	00001540	RG	01
\$\$BASS\$CVT_OUT_P_F	00001520	RG	01	\$\$BASS\$NUM_H	00001548	RG	01
\$\$BASS\$CVT_OUT_P_G	00001528	RG	01	\$\$BASS\$NUM_L	000008E0	RG	01
\$\$BASS\$CVT_T_P	00001480	RG	01	\$\$BASS\$NUM_P	00001550	RG	01
\$\$BASS\$DELETE	00000AE0	RG	01	\$\$BASS\$ON_ERR_BK	00000960	RG	01
\$\$BASS\$SCALE_D_R1	00000880	RG	01	\$\$BASS\$ON_ERR_Z	00000958	RG	01
\$\$BASS\$EDIT	000008A8	RG	01	\$\$BASS\$OPEN	00000A80	RG	01
\$\$BASS\$END_DEF_R8	00000940	RG	01	\$\$BASS\$OUT_D_V_B	00000A38	RG	01
\$\$BASS\$END_DFS_R8	00000948	RG	01	\$\$BASS\$OUT_D_V_C	00000A40	RG	01
\$\$BASS\$END_GSB_R8	00000950	RG	01	\$\$BASS\$OUT_D_V_S	00000A30	RG	01
\$\$BASS\$END_R8	00000938	RG	01	\$\$BASS\$OUT_F_V_B	00000A20	RG	01
\$\$BASS\$ERL	00000980	RG	01	\$\$BASS\$OUT_F_V_C	00000A28	RG	01

VM\$VECTOR
Symbol table

- Define entry vectors for VMSRTL

D 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1

Page 72
(35)

\$\$BASSOUT_F_V_S	00000A18	RG	01	\$\$COBSACC_TIME	00001400	RG	01
\$\$BASSOUT_G_V_B	00001590	RG	01	\$\$COBSADDI	00001330	RG	01
\$\$BASSOUT_G_V_C	00001598	RG	01	\$\$COBS\$CMPI	00001358	RG	01
\$\$BASSOUT_G_V_S	00001588	RG	01	\$\$COBS\$CVTDI_R7	00001360	RG	01
\$\$BASSOUT_H_V_B	000015A8	RG	01	\$\$COBS\$CVTFI_R7	00001368	RG	01
\$\$BASSOUT_H_V_C	000015B0	RG	01	\$\$COBS\$CVTID_R7	00001370	RG	01
\$\$BASSOUT_H_V_S	000015A0	RG	01	\$\$COBS\$CVTIF_R7	00001378	RG	01
\$\$BASSOUT_L_V_B	00000A08	RG	01	\$\$COBS\$CVTIL_R8	00001380	RG	01
\$\$BASSOUT_L_V_C	00000A10	RG	01	\$\$COBS\$CVTIP_R9	00001388	RG	01
\$\$BASSOUT_L_V_S	00000A00	RG	01	\$\$COBS\$CVTIQ_R8	00001390	RG	01
\$\$BASSOUT_P_DX_B	000015C0	RG	01	\$\$COBS\$CVTIW_R8	00001398	RG	01
\$\$BASSOUT_P_DX_C	000015C8	RG	01	\$\$COBS\$CVTLI_R8	000013A0	RG	01
\$\$BASSOUT_P_DX_S	000015B8	RG	01	\$\$COBS\$CVTPI_R9	000013A8	RG	01
\$\$BASSOUT_T_DX_B	00000A50	RG	01	\$\$COBS\$CVTPQ_R9	00001438	RG	01
\$\$BASSOUT_T_DX_C	00000A58	RG	01	\$\$COBS\$CVTQI_R8	000013B0	RG	01
\$\$BASSOUT_T_DX_S	00000A48	RG	01	\$\$COBS\$CVTQP_R9	00001440	RG	01
\$\$BASSPOP_ERR	00000B68	RG	01	\$\$COBS\$CVTRIC_R8	000013B8	RG	01
\$\$BASSPRINT	000009C0	RG	01	\$\$COBS\$CVTRIP_R9	000013C0	RG	01
\$\$BASSPRINT_USING	000009C8	RG	01	\$\$COBS\$CVTRIQ_R8	000013C8	RG	01
\$\$BASSPUSH_ERR	00000B60	RG	01	\$\$COBS\$CVTRIW_R8	000013D0	RG	01
\$\$BASSPUT	00000AA8	RG	01	\$\$COBS\$CVTRPQ_R9	00001448	RG	01
\$\$BASSPUT_COUNT	00000AB8	RG	01	\$\$COBS\$CVTRQP_R9	00001450	RG	01
\$\$BASSPUT_RECORD	00000AB0	RG	01	\$\$COBS\$CVTTI_R8	000013D8	RG	01
\$\$BASSPUT_REC_CNT	00000AC0	RG	01	\$\$COBS\$CVTWI_R8	000013E0	RG	01
\$\$BASSRCTRLC	00001640	RG	01	\$\$COBS\$DISPLAY	00001410	RG	01
\$\$BASSREAD	000009B8	RG	01	\$\$COBS\$DISP_NO_ADV	00001418	RG	01
\$\$BASSRECOUNT	00000B30	RG	01	\$\$COBS\$DIVI	00001348	RG	01
\$\$BASSRESTORE	00000AF8	RG	01	\$\$COBS\$DIVI_OSE	00001350	RG	01
\$\$BASSRESTORE_DAT	00000B28	RG	01	\$\$COBS\$DIVQ_R8	00001420	RG	01
\$\$BASSRESTORE_KEY	00000B00	RG	01	\$\$COBS\$ERROR	00001328	RG	01
\$\$BASSRESUME	00000968	RG	01	\$\$COBS\$HANDLER	00001318	RG	01
\$\$BASSRESUME_Z	00000970	RG	01	\$\$COBS\$IOEXCEPTION	00001320	RG	01
\$\$BASSRSET	00000898	RG	01	\$\$COBS\$MULI	00001340	RG	01
\$\$BASSRSET_R	000008A0	RG	01	\$\$COBS\$MULQ_R8	00001428	RG	01
\$\$BASSSCALE_D_R1	00000878	RG	01	\$\$COBS\$PAUSE	00001430	RG	01
\$\$BASS\$SCRATCH	00000B08	RG	01	\$\$COBS\$SUBI	00001338	RG	01
\$\$BASS\$STATUS	00000B38	RG	01	\$\$FOR\$SCB_GET	00000620	RG	01
\$\$BASS\$STR_D	000008C0	RG	01	\$\$FOR\$SCB_POP	00000610	RG	01
\$\$BASS\$STR_F	000008B8	RG	01	\$\$FOR\$SCB_PUSH	00000608	RG	01
\$\$BASS\$STR_G	00001570	RG	01	\$\$FOR\$SCB_RET	00000618	RG	01
\$\$BASS\$STR_H	00001578	RG	01	\$\$FOR\$SERRSNS_SAV	00000628	RG	01
\$\$BASS\$STR_L	000008C8	RG	01	\$\$FOR\$SFP_MATCH	00000728	RG	01
\$\$BASS\$STR_P	00001580	RG	01	\$\$FOR\$BACKSPACE	00000180	RG	01
\$\$BASSUNLOCK	00000B10	RG	01	\$\$FOR\$CLOSE	00000000	RG	01
\$\$BASSUPDATE	00000AE8	RG	01	\$\$FOR\$CNV_IN_DEFG	00000200	RG	01
\$\$BASSUPDATE_COUN	00000AF0	RG	01	\$\$FOR\$CNV_IN_I	00000210	RG	01
\$\$BASS\$VAL_D	00000910	RG	01	\$\$FOR\$CNV_IN_L	00000218	RG	01
\$\$BASS\$VAL_F	00000908	RG	01	\$\$FOR\$CNV_IN_O	00000220	RG	01
\$\$BASS\$VAL_G	000015E8	RG	01	\$\$FOR\$CNV_IN_Z	00000228	RG	01
\$\$BASS\$VAL_H	000015F0	RG	01	\$\$FOR\$CNV_OUT_D	000001A8	RG	01
\$\$BASS\$VAL_L	00000900	RG	01	\$\$FOR\$CNV_OUT_E	000001B0	RG	01
\$\$BASS\$VAL_P	000015F8	RG	01	\$\$FOR\$CNV_OUT_F	000001B8	RG	01
\$\$BASS\$WAIT	00000CA0	RG	01	\$\$FOR\$CNV_OUT_G	000001C0	RG	01
\$\$COBS\$ACCEPT	00001408	RG	01	\$\$FOR\$CNV_OUT_I	00000188	RG	01
\$\$COBS\$ACC_DATE	000013E8	RG	01	\$\$FOR\$CNV_OUT_L	00000190	RG	01
\$\$COBS\$ACC_DAY	000013F0	RG	01	\$\$FOR\$CNV_OUT_O	00000198	RG	01
\$\$COBS\$ACC_DAYWEEK	000013F8	RG	01	\$\$FOR\$CNV_OUT_Z	000001A0	RG	01

VMSSVECTOR
Symbol table

- Define entry vectors for VMSRTL E 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 73
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (35)

\$\$FOR\$CVT_D_TD	000001A8	RG	01	\$\$FOR\$IO_X_DA	00000170	RG	01
\$\$FOR\$CVT_D_TE	000001B0	RG	01	\$\$FOR\$IO_X_NL	00000748	RG	01
\$\$FOR\$CVT_D_TF	000001B8	RG	01	\$\$FOR\$IO_X_SB	00000740	RG	01
\$\$FOR\$CVT_D_TG	000001C0	RG	01	\$\$FOR\$IO_X_SE	00000750	RG	01
\$\$FOR\$CVT_G_TD	00000640	RG	01	\$\$FOR\$OPEN	00000178	RG	01
\$\$FOR\$CVT_G_TE	00000648	RG	01	\$\$FOR\$PAUSE	00000248	RG	01
\$\$FOR\$CVT_G_TF	00000650	RG	01	\$\$FOR\$RAB	00000788	RG	01
\$\$FOR\$CVT_G_TG	00000658	RG	01	\$\$FOR\$READ_DF	00000038	RG	01
\$\$FOR\$CVT_H_TD	00000668	RG	01	\$\$FOR\$READ_DO	00000040	RG	01
\$\$FOR\$CVT_H_TE	00000670	RG	01	\$\$FOR\$READ_DU	00000048	RG	01
\$\$FOR\$CVT_H_TF	00000678	RG	01	\$\$FOR\$READ_IF	000006C8	RG	01
\$\$FOR\$CVT_H_TG	00000680	RG	01	\$\$FOR\$READ_IO	000006D0	RG	01
\$\$FOR\$DECODE_MF	00000008	RG	01	\$\$FOR\$READ_KF	00000028	RG	01
\$\$FOR\$DECODE_MO	00000010	RG	01	\$\$FOR\$READ_KO	00000030	RG	01
\$\$FOR\$DEF_FICE	000001C8	RG	01	\$\$FOR\$READ_KU	00000708	RG	01
\$\$FOR\$DEF_FILE_W	000001D0	RG	01	\$\$FOR\$READ_SF	00000050	RG	01
\$\$FOR\$DELETE	000006E8	RG	01	\$\$FOR\$READ_SL	00000058	RG	01
\$\$FOR\$DELETE_D	000006F0	RG	01	\$\$FOR\$READ_SN	00000730	RG	01
\$\$FOR\$ENCODE_MF	00000018	RG	01	\$\$FOR\$READ_SO	00000060	RG	01
\$\$FOR\$ENCODE_MO	00000020	RG	01	\$\$FOR\$READ_SU	00000068	RG	01
\$\$FOR\$ENDFILE	000001D8	RG	01	\$\$FOR\$REWIND	00000250	RG	01
\$\$FOR\$ERRSNS	000001E0	RG	01	\$\$FOR\$REWRITE_SF	000006B0	RG	01
\$\$FOR\$ERRSNS_W	000001E8	RG	01	\$\$FOR\$REWRITE_SO	000006B8	RG	01
\$\$FOR\$EXIT	000001F0	RG	01	\$\$FOR\$REWRITE_SU	000006C0	RG	01
\$\$FOR\$EXIT_W	000001F8	RG	01	\$\$FOR\$SECNDS	00000258	RG	01
\$\$FOR\$FIND	00000208	RG	01	\$\$FOR\$STOP	00000260	RG	01
\$\$FOR\$INI_DES1_R2	00000230	RG	01	\$\$FOR\$UNLOCK	00000700	RG	01
\$\$FOR\$INI_DES2_R3	00000238	RG	01	\$\$FOR\$WRITE_DF	00000070	RG	01
\$\$FOR\$INI_DESC_R6	00000240	RG	01	\$\$FOR\$WRITE_DO	00000078	RG	01
\$\$FOR\$INQUIRE	000006F8	RG	01	\$\$FOR\$WRITE_DU	00000080	RG	01
\$\$FOR\$IO_B_R	000000E0	RG	01	\$\$FOR\$WRITE_IF	000006D8	RG	01
\$\$FOR\$IO_B_V	000000E8	RG	01	\$\$FOR\$WRITE_IO	000006E0	RG	01
\$\$FOR\$IO_DC_R	00000128	RG	01	\$\$FOR\$WRITE_SF	00000088	RG	01
\$\$FOR\$IO_DC_V	00000630	RG	01	\$\$FOR\$WRITE_SL	00000090	RG	01
\$\$FOR\$IO_D_R	000000C0	RG	01	\$\$FOR\$WRITE_SN	00000738	RG	01
\$\$FOR\$IO_D_V	000000C8	RG	01	\$\$FOR\$WRITE_SO	00000098	RG	01
\$\$FOR\$IO_END	000000A8	RG	01	\$\$FOR\$WRITE_SU	000000A0	RG	01
\$\$FOR\$IO_FC_R	00000140	RG	01	\$\$LIB\$ANALYZE_SDESC	00001480	RG	01
\$\$FOR\$IO_FC_V	00000148	RG	01	\$\$LIB\$ANALYZE_SDESC_R2	00001488	RG	01
\$\$FOR\$IO_F_R	000000B0	RG	01	\$\$LIB\$AST_IN_PROG	000004B0	RG	01
\$\$FOR\$IO_F_V	000000B8	RG	01	\$\$LIB\$ATTACH	00000770	RG	01
\$\$FOR\$IO_GC_R	00000130	RG	01	\$\$LIB\$CRC	000004B8	RG	01
\$\$FOR\$IO_GC_V	00000638	RG	01	\$\$LIB\$CRC_TABLE	000004C0	RG	01
\$\$FOR\$IO_G_R	00000108	RG	01	\$\$LIB\$DEC_OVER	000004C8	RG	01
\$\$FOR\$IO_G_V	00000110	RG	01	\$\$LIB\$ESTABLISH	000004D0	RG	01
\$\$FOR\$IO_H_R	00000118	RG	01	\$\$LIB\$EXTV	000004D8	RG	01
\$\$FOR\$IO_H_V	00000120	RG	01	\$\$LIB\$EXTZV	000004E0	RG	01
\$\$FOR\$IO_LD_R	00000150	RG	01	\$\$LIB\$FFC	000004E8	RG	01
\$\$FOR\$IO_LU_V	00000158	RG	01	\$\$LIB\$FFS	000004F0	RG	01
\$\$FOR\$IO_L_R	000000D0	RG	01	\$\$LIB\$FILE_SCAN	000014A0	RG	01
\$\$FOR\$IO_L_V	000000D8	RG	01	\$\$LIB\$FIND_FILE	000014A8	RG	01
\$\$FOR\$IO_T_DS	000000F0	RG	01	\$\$LIB\$FIXUP_FLT	000004F8	RG	01
\$\$FOR\$IO_T_V_DS	00000138	RG	01	\$\$LIB\$FLT_UNDER	00000500	RG	01
\$\$FOR\$IO_WD_R	00000160	RG	01	\$\$LIB\$FREE_EF	00001468	RG	01
\$\$FOR\$IO_WU_V	00000168	RG	01	\$\$LIB\$FREE_LUN	00001458	RG	01
\$\$FOR\$IO_W_R	000000F8	RG	01	\$\$LIB\$FREE_VM	000005F0	RG	01
\$\$FOR\$IO_W_V	00000100	RG	01	\$\$LIB\$GET_COMMAND	00000510	RG	01

VMSSVECTOR
Symbol table

- Define entry vectors for VMSRTL

F 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1

Page 74
(35)

\$\$LIB\$GET_EF	00001470	RG	01	\$\$MTH\$ATAND2	00000E28	RG	01
\$\$LIB\$GET_INPUT	00000508	RG	01	\$\$MTH\$ATAND_R4	00000E30	RG	01
\$\$LIB\$GET_LUN	00001460	RG	01	\$\$MTH\$ATANH	00000ED0	RG	01
\$\$LIB\$GET_OPCODE	00000780	RG	01	\$\$MTH\$ATAN_R4	000002B8	RG	01
\$\$LIB\$GET_VM	000005F8	RG	01	\$\$MTH\$CABS	00000438	RG	01
\$\$LIB\$INDEX	00000518	RG	01	\$\$MTH\$CCOS	00000458	RG	01
\$\$LIB\$INSV	00000520	RG	01	\$\$MTH\$CEXP	00000440	RG	01
\$\$LIB\$INT_OVER	00000528	RG	01	\$\$MTH\$CLOG	00000448	RG	01
\$\$LIB\$LOC	00000530	RG	01	\$\$MTH\$COS	00000368	RG	01
\$\$LIB\$MATCHC	00000538	RG	01	\$\$MTH\$COSD	00000E90	RG	01
\$\$LIB\$MATCH_COND	00000540	RG	01	\$\$MTH\$COSD_R4	00000E98	RG	01
\$\$LIB\$MOVTC	00000548	RG	01	\$\$MTH\$COSH	00000450	RG	01
\$\$LIB\$MOVTUC	00000550	RG	01	\$\$MTH\$COS_R4	00000370	RG	01
\$\$LIB\$PUT_OUTPUT	00000558	RG	01	\$\$MTH\$CSIN	00000460	RG	01
\$\$LIB\$RESERVE_EF	00001478	RG	01	\$\$MTH\$CSQRT	00000468	RG	01
\$\$LIB\$REVERT	00000560	RG	01	\$\$MTH\$DACOS	000002C0	RG	01
\$\$LIB\$SCANC	00000568	RG	01	\$\$MTH\$DACOSD	00000E38	RG	01
\$\$LIB\$SCOPY_DXDX	00000570	RG	01	\$\$MTH\$DACOSD_R7	00000E40	RG	01
\$\$LIB\$SCOPY_DXDX6	00000578	RG	01	\$\$MTH\$DACOS_R7	000002C8	RG	01
\$\$LIB\$SCOPY_R_DX	00000580	RG	01	\$\$MTH\$DACOS_R9	000002C8	RG	01
\$\$LIB\$SCOPY_R_DX6	00000588	RG	01	\$\$MTH\$DASIN	000002D0	RG	01
\$\$LIB\$SFREET_DD	000005A0	RG	01	\$\$MTH\$DASIND	00000E48	RG	01
\$\$LIB\$SFREET1_DD6	000005A8	RG	01	\$\$MTH\$DASIND_R7	00000E50	RG	01
\$\$LIB\$SFREEN_DD	000005B0	RG	01	\$\$MTH\$DASIN_R7	000002D8	RG	01
\$\$LIB\$SFREEN_DD6	000005B8	RG	01	\$\$MTH\$DASIN_R9	000002D8	RG	01
\$\$LIB\$SGET1_DD	00000590	RG	01	\$\$MTH\$DATAN	000002E0	RG	01
\$\$LIB\$SGET1_DD_R6	00000598	RG	01	\$\$MTH\$DATAN2	000002E8	RG	01
\$\$LIB\$SHOW_VM	00000600	RG	01	\$\$MTH\$DATAND	00000E58	RG	01
\$\$LIB\$SIGNL	000005C8	RG	01	\$\$MTH\$DATAND2	00000E60	RG	01
\$\$LIB\$SIG_TO_RET	000005D8	RG	01	\$\$MTH\$DATAND_R7	00000E68	RG	01
\$\$LIB\$SKPC	000005E0	RG	01	\$\$MTH\$DATANH	00000ED8	RG	01
\$\$LIB\$SPANC	000005E8	RG	01	\$\$MTH\$DATAN_R7	000002F0	RG	01
\$\$LIB\$SPAWN	00000778	RG	01	\$\$MTH\$DCOS	00000328	RG	01
\$\$LIB\$STAT_VM	000005C0	RG	01	\$\$MTH\$DCOSD	00000E70	RG	01
\$\$LIB\$STOP	000005D0	RG	01	\$\$MTH\$DCOSD_R7	00000E78	RG	01
\$\$LIB\$TPARSE	00000B58	RG	01	\$\$MTH\$DCOSH	00000470	RG	01
\$\$MTH\$SAB ALOG_V	00000F48	RG	01	\$\$MTH\$DCOS_R7	00000330	RG	01
\$\$MTH\$SAB ATAN_V	00000F50	RG	01	\$\$MTH\$DEXP	000002F8	RG	01
\$\$MTH\$ACOS	00000268	RG	01	\$\$MTH\$DEXP_R6	00000300	RG	01
\$\$MTH\$ACOSD	00000E00	RG	01	\$\$MTH\$DEXP_R7	00000300	RG	01
\$\$MTH\$ACOSD_R4	00000E08	RG	01	\$\$MTH\$DLOG	00000308	RG	01
\$\$MTH\$ACOS_R4	00000270	RG	01	\$\$MTH\$DLOG10	00000310	RG	01
\$\$MTH\$ACOS_R5	00000270	RG	01	\$\$MTH\$DLOG10_R8	00000318	RG	01
\$\$MTH\$ALOG	00000278	RG	01	\$\$MTH\$DLOG2	00000F28	RG	01
\$\$MTH\$ALOG10	00000280	RG	01	\$\$MTH\$DLOG_R8	00000320	RG	01
\$\$MTH\$ALOG10_R5	00000288	RG	01	\$\$MTH\$DSIN	00000338	RG	01
\$\$MTH\$ALOG2	00000F20	RG	01	\$\$MTH\$DSINCOS	00000F00	RG	01
\$\$MTH\$ALOG_R5	00000290	RG	01	\$\$MTH\$DSINCOSD	00000F10	RG	01
\$\$MTH\$AL_4_OV_PI_V	00000F40	RG	01	\$\$MTH\$DSINCOSD_R7	00000F18	RG	01
\$\$MTH\$ASIN	00000298	RG	01	\$\$MTH\$DSINCOS_R7	00000F08	RG	01
\$\$MTH\$ASIND	00000E10	RG	01	\$\$MTH\$DSIND	00000E80	RG	01
\$\$MTH\$ASIND_R4	00000E18	RG	01	\$\$MTH\$DSIND_R7	00000E88	RG	01
\$\$MTH\$ASIN_R4	000002A0	RG	01	\$\$MTH\$DSINH	00000478	RG	01
\$\$MTH\$ASIN_R5	000002A0	RG	01	\$\$MTH\$DSIN_R7	00000340	RG	01
\$\$MTH\$ATAN	000002A8	RG	01	\$\$MTH\$DSQRT	00000348	RG	01
\$\$MTH\$ATAN2	000002B0	RG	01	\$\$MTH\$DSQRT_R5	00000350	RG	01
\$\$MTH\$ATAND	00000E20	RG	01	\$\$MTH\$DTAN	00000480	RG	01

VMSSVECTOR
Symbol table

- Define entry vectors for VMSRTL G 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 75
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (35)

\$\$MTH\$DTAND	00000EB0	RG	01	\$\$OT\$\$\$FREE1-DD	00000418	RG	01
\$\$MTH\$DTAND_R7	00000EB8	RG	01	\$\$OT\$\$\$FREE1-DD6	00000420	RG	01
\$\$MTH\$DTANH	00000488	RG	01	\$\$OT\$\$\$FREE1-DD	00000428	RG	01
\$\$MTH\$DTAN_R7	00000710	RG	01	\$\$OT\$\$\$FREE1-DD6	00000430	RG	01
\$\$MTH\$EXP	00000358	RG	01	\$\$OT\$\$\$GET1-DD	00000408	RG	01
\$\$MTH\$EXP_R4	00000360	RG	01	\$\$OT\$\$\$GET1-DD_R6	00000410	RG	01
\$\$MTH\$RANDOM	00000490	RG	01	\$\$STR\$ANALYZE_SDESC	00001490	RG	01
\$\$MTH\$SIN	00000378	RG	01	\$\$STR\$ANALYZE_SDESC_R1	00001498	RG	01
\$\$MTH\$SINCOS	00000EE0	RG	01	\$\$STR\$APPEND	00000C60	RG	01
\$\$MTH\$SINCOSD	00000EF0	RG	01	\$\$STR\$COMPARE	00000C68	RG	01
\$\$MTH\$SINCOSD_R5	00000EF8	RG	01	\$\$STR\$COMPARE_EQL	00000C70	RG	01
\$\$MTH\$SINCOS_R5	00000EE8	RG	01	\$\$STR\$CONCAT	00000800	RG	01
\$\$MTH\$SIND	00000EA0	RG	01	\$\$STR\$COPY_DX	00000808	RG	01
\$\$MTH\$SIND_R4	00000EA8	RG	01	\$\$STR\$COPY_DX_R8	00000C10	RG	01
\$\$MTH\$SINH	00000498	RG	01	\$\$STR\$COPY_R	00000810	RG	01
\$\$MTH\$SIN_R4	00000380	RG	01	\$\$STR\$COPY_R_R8	00000C18	RG	01
\$\$MTH\$SQRT	00000388	RG	01	\$\$STR\$DUPL_CHAR	00000850	RG	01
\$\$MTH\$SQRT_R2	00000390	RG	01	\$\$STR\$DUPL_CHARR8	00000C20	RG	01
\$\$MTH\$SQRT_R3	00000720	RG	01	\$\$STR\$FREE1_DX	00000818	RG	01
\$\$MTH\$TAN	000004A0	RG	01	\$\$STR\$FREE1_DX_R4	00000C28	RG	01
\$\$MTH\$TAND	00000EC0	RG	01	\$\$STR\$GET1_DX	00000820	RG	01
\$\$MTH\$TAND_R4	00000EC8	RG	01	\$\$STR\$GET1_DX_R4	00000C30	RG	01
\$\$MTH\$TAND_R5	00000F38	RG	01	\$\$STR\$LEFT	00000828	RG	01
\$\$MTH\$TANH	000004A8	RG	01	\$\$STR\$LEFT_R8	00000C38	RG	01
\$\$MTH\$TAN_R4	00000718	RG	01	\$\$STR\$LEN_EXTR	00000830	RG	01
\$\$MTH\$TAN_R5	00000F30	RG	01	\$\$STR\$LEN_EXTR_R8	00000C40	RG	01
\$\$OT\$\$\$CVT_D_T_R8	00000790	RG	01	\$\$STR\$POSITION	00000840	RG	01
\$\$OT\$\$\$CVT_G_T_R8	00000798	RG	01	\$\$STR\$POSITION_R6	00000C48	RG	01
\$\$OT\$\$\$CVT_H_T_R8	000007A0	RG	01	\$\$STR\$POS_EXTR	00000838	RG	01
\$\$OT\$\$\$CVT_L_TB	00000758	RG	01	\$\$STR\$POS_EXTR_R8	00000C50	RG	01
\$\$OT\$\$\$CVT_L_TI	00000690	RG	01	\$\$STR\$PREFIX	00000C78	RG	01
\$\$OT\$\$\$CVT_L_TL	000006A8	RG	01	\$\$STR\$REPLACE	00000C80	RG	01
\$\$OT\$\$\$CVT_L_TO	00000698	RG	01	\$\$STR\$REPLACE_R8	00000C88	RG	01
\$\$OT\$\$\$CVT_L_TZ	000006A0	RG	01	\$\$STR\$RIGHT	00000848	RG	01
\$\$OT\$\$\$CVT_TB_L	00000760	RG	01	\$\$STR\$RIGHT_R8	00000C58	RG	01
\$\$OT\$\$\$CVT_TI_L	00000210	RG	01	\$\$STR\$TRANSCATE	00000C90	RG	01
\$\$OT\$\$\$CVT_TL_L	00000218	RG	01	\$\$STR\$TRIM	00000858	RG	01
\$\$OT\$\$\$CVT_TO_L	00000220	RG	01	\$\$STR\$UPCASE	00000C98	RG	01
\$\$OT\$\$\$CVT_TZ_L	00000228	RG	01	BASS\$BLNK_LINE	*****	X	01
\$\$OT\$\$\$CVT_T_D	00000200	RG	01	BASS\$CB_GET	*****	X	01
\$\$OT\$\$\$CVT_T_F	00000768	RG	01	BASS\$CB_POP	*****	X	01
\$\$OT\$\$\$CVT_T_G	00000660	RG	01	BASS\$CB_PUSH	*****	X	01
\$\$OT\$\$\$CVT_T_H	00000688	RG	01	BASS\$CLOSE_ALL	*****	X	01
\$\$OT\$\$\$DIVC	00000398	RG	01	BASS\$CTRLC_INIT	*****	X	01
\$\$OT\$\$\$POWCJ	000003A0	RG	01	BASS\$ERR_INIT	*****	X	01
\$\$OT\$\$\$POWDD	000003A8	RG	01	BASS\$FORMAT_INT	*****	X	01
\$\$OT\$\$\$POWDJ	000003C0	RG	01	BASS\$HANDLER	*****	X	01
\$\$OT\$\$\$POWDR	000003B0	RG	01	BASS\$NEXT_LUN	*****	X	01
\$\$OT\$\$\$POWII	000003C8	RG	01	BASS\$OPEN_ZERO	*****	X	01
\$\$OT\$\$\$POWJJ	000003D0	RG	01	BASS\$RECOO_INIT	*****	X	01
\$\$OT\$\$\$POWRD	000003B8	RG	01	BASS\$REC_WSL1	*****	X	01
\$\$OT\$\$\$POWRJ	000003D8	RG	01	BASS\$SCALE_L_R1	*****	X	01
\$\$OT\$\$\$POWRR	000003E0	RG	01	BASS\$SCALE_RT	*****	X	01
\$\$OT\$\$\$SCOPY_DXDX	000003E8	RG	01	BASS\$SIGNAL	*****	X	01
\$\$OT\$\$\$SCOPY_DXDX6	000003F0	RG	01	BASS\$SIGNAL_IO	*****	X	01
\$\$OT\$\$\$SCOPY_R_DX	000003F8	RG	01	BASS\$STATU_INIT	*****	X	01
\$\$OT\$\$\$SCOPY_R_DX6	00000400	RG	01	BASS\$STOP	*****	X	01

VMSSVECTOR
Symbol table

- Define entry vectors for VMSRTL H 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 76
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (35)

BASS\$STOP_IO	*****	X	01	BASS\$INIT_R8	*****	X	01
BASS\$STOP-RMS	*****	X	01	BASS\$INPUT	*****	X	01
BASS\$UDF_RL1	*****	X	01	BASS\$INPUT_LINE	*****	X	01
BASS\$UDF_WL1	*****	X	01	BASS\$INSTR	*****	X	01
BASS\$ANSI_INPUT	*****	X	01	BASS\$IN_B_R	*****	X	01
BASS\$ANSI_IO_END	*****	X	01	BASS\$IN_D_R	*****	X	01
BASS\$BUF\$IZ	*****	X	01	BASS\$IN_F_R	*****	X	01
BASS\$CANTYPAHEAD	*****	X	01	BASS\$IN_G_R	*****	X	01
BASS\$CCPOS	*****	X	01	BASS\$IN_H_R	*****	X	01
BASS\$CHR	*****	X	01	BASS\$IN_L_R	*****	X	01
BASS\$CLOSE	*****	X	01	BASS\$IN_P_DX	*****	X	01
BASS\$CMPD_APP	*****	X	01	BASS\$IN_T_DX	*****	X	01
BASS\$CMPF_APP	*****	X	01	BASS\$IN_W_R	*****	X	01
BASS\$CMPI_APP	*****	X	01	BASS\$IO_END	*****	X	01
BASS\$CMPI_APP	*****	X	01	BASS\$INPUT	*****	X	01
BASS\$CTRLC	*****	X	01	BASS\$MAT_INPUT	*****	X	01
BASS\$CVT_OUT_D_E	*****	X	01	BASS\$MAT_LINPUT	*****	X	01
BASS\$CVT_OUT_D_F	*****	X	01	BASS\$MAT_PRINT	*****	X	01
BASS\$CVT_OUT_D_G	*****	X	01	BASS\$MAT_READ	*****	X	01
BASS\$CVT_OUT_F_E	*****	X	01	BASS\$NUMT_D	*****	X	01
BASS\$CVT_OUT_F_F	*****	X	01	BASS\$NUM1_F	*****	X	01
BASS\$CVT_OUT_G_E	*****	X	01	BASS\$NUM1_G	*****	X	01
BASS\$CVT_OUT_G_F	*****	X	01	BASS\$NUM1_H	*****	X	01
BASS\$CVT_OUT_G_G	*****	X	01	BASS\$NUM1_L	*****	X	01
BASS\$CVT_OUT_H_E	*****	X	01	BASS\$NUM1_P	*****	X	01
BASS\$CVT_OUT_H_F	*****	X	01	BASS\$NUM_B	*****	X	01
BASS\$CVT_OUT_H_G	*****	X	01	BASS\$NUM_F	*****	X	01
BASS\$CVT_OUT_P_E	*****	X	01	BASS\$NUM_G	*****	X	01
BASS\$CVT_OUT_P_F	*****	X	01	BASS\$NUM_H	*****	X	01
BASS\$CVT_OUT_P_G	*****	X	01	BASS\$NUM_L	*****	X	01
BASS\$CVT_T_P	*****	X	01	BASS\$NUM_P	*****	X	01
BASS\$DELETE	*****	X	01	BASS\$ON_ERR_BK	*****	X	01
BASS\$DS\$CALE_D_R1	*****	X	01	BASS\$ON_ERR_Z	*****	X	01
BASS\$EDIT	*****	X	01	BASS\$OPEN	*****	X	01
BASS\$END_DEF_R8	*****	X	01	BASS\$OUT_D_V_B	*****	X	01
BASS\$END_DFS_R8	*****	X	01	BASS\$OUT_D_V_C	*****	X	01
BASS\$END_GSB_R8	*****	X	01	BASS\$OUT_D_V_S	*****	X	01
BASS\$END_R8	*****	X	01	BASS\$OUT_F_V_B	*****	X	01
BASS\$ERL	*****	X	01	BASS\$OUT_F_V_C	*****	X	01
BASS\$ERN	*****	X	01	BASS\$OUT_F_V_S	*****	X	01
BASS\$ERR	*****	X	01	BASS\$OUT_G_V_B	*****	X	01
BASS\$ERROR	*****	X	01	BASS\$OUT_G_V_C	*****	X	01
BASS\$ERT	*****	X	01	BASS\$OUT_G_V_S	*****	X	01
BASS\$FIND	*****	X	01	BASS\$OUT_H_V_B	*****	X	01
BASS\$FIND_KEY	*****	X	01	BASS\$OUT_H_V_C	*****	X	01
BASS\$FIND_RECORD	*****	X	01	BASS\$OUT_H_V_S	*****	X	01
BASS\$FIND_RFA	*****	X	01	BASS\$OUT_L_V_B	*****	X	01
BASS\$FREE	*****	X	01	BASS\$OUT_L_V_C	*****	X	01
BASS\$GET	*****	X	01	BASS\$OUT_L_V_S	*****	X	01
BASS\$GETRFA	*****	X	01	BASS\$OUT_P_DX_B	*****	X	01
BASS\$GET_KEY	*****	X	01	BASS\$OUT_P_DX_C	*****	X	01
BASS\$GET_RECORD	*****	X	01	BASS\$OUT_P_DX_S	*****	X	01
BASS\$GET_RFA	*****	X	01	BASS\$OUT_T_DX_B	*****	X	01
BASS\$HANDLER	*****	X	01	BASS\$OUT_T_DX_C	*****	X	01
BASS\$INIT_DEF_R8	*****	X	01	BASS\$OUT_T_DX_S	*****	X	01
BASS\$INIT_DFS_R8	*****	X	01	BASS\$POP_ERR	*****	X	01
BASS\$INIT_GOSOB	*****	X	01	BASS\$PRINT	*****	X	01

[illegible]

[illegible]

VMSSVECTOR
Symbol table

- Define entry vectors for VMSRTL K 13

16-SEP-1984 02:15:59 VAX/VMS Macro V04-00 Page 79
6-SEP-1984 11:48:04 [VMSRTL.SRC]VMSVECTOR.MAR;1 (35)

LIBSSGET1_DD	*****	X	01	MTH\$DATAND_R7	*****	X	01
LIBSSGET1_DD_R6	*****	X	01	MTH\$DATANH	*****	X	01
LIBSSHOW_VM	*****	X	01	MTH\$DATAN_R7	*****	X	01
LIBSSIGNAL	*****	X	01	MTH\$DCOS	*****	X	01
LIBSSIG_TO_RET	*****	X	01	MTH\$DCOSD	*****	X	01
LIBSSKPC	*****	X	01	MTH\$DCOSD_R7	*****	X	01
LIBSSPANC	*****	X	01	MTH\$DCOSH	*****	X	01
LIBSSPAWN	*****	X	01	MTH\$DCOS_R7	*****	X	01
LIBSSSTAT_VM	*****	X	01	MTH\$DEXP	*****	X	01
LIBSSSTOP	*****	X	01	MTH\$DEXP_R6	*****	X	01
LIBSTPARSE	*****	X	01	MTH\$DLOG	*****	X	01
MTH\$SAB ALOG	00001650	R	01	MTH\$DLOG10	*****	X	01
MTH\$SAB ATAN	00001750	R	01	MTH\$DLOG10_R8	*****	X	01
MTH\$ACOS	*****	X	01	MTH\$DLOG2	*****	X	01
MTH\$ACOSD	*****	X	01	MTH\$DLOG_R8	*****	X	01
MTH\$ACOSD_R4	*****	X	01	MTH\$DSIN	*****	X	01
MTH\$ACOS_R4	*****	X	01	MTH\$DSINCOS	*****	X	01
MTH\$ALOG	*****	X	01	MTH\$DSINCOSD	*****	X	01
MTH\$ALOG10	*****	X	01	MTH\$DSINCOSD_R7	*****	X	01
MTH\$ALOG10_R5	*****	X	01	MTH\$DSINCOS_R7	*****	X	01
MTH\$ALOG2	*****	X	01	MTH\$DSIND	*****	X	01
MTH\$ALOG_R5	*****	X	01	MTH\$DSIND_R7	*****	X	01
MTH\$AL_4_OV_P1	*****	X	01	MTH\$DSINH	*****	X	01
MTH\$ASIN	*****	X	01	MTH\$DSIN_R7	*****	X	01
MTH\$ASIND	*****	X	01	MTH\$DSQRT	*****	X	01
MTH\$ASIND_R4	*****	X	01	MTH\$DSQRT_R5	*****	X	01
MTH\$ASIN_R4	*****	X	01	MTH\$DTAN	*****	X	01
MTH\$ATAN	*****	X	01	MTH\$DTAND	*****	X	01
MTH\$ATAN2	*****	X	01	MTH\$DTAND_R7	*****	X	01
MTH\$ATAND	*****	X	01	MTH\$DTANH	*****	X	01
MTH\$ATAND2	*****	X	01	MTH\$DTAN_R7	*****	X	01
MTH\$ATAND_R4	*****	X	01	MTH\$EXP	*****	X	01
MTH\$ATANH	*****	X	01	MTH\$EXP_R4	*****	X	01
MTH\$ATAN_R4	*****	X	01	MTH\$RANDOM	*****	X	01
MTH\$CABS	*****	X	01	MTH\$SIN	*****	X	01
MTH\$CCOS	*****	X	01	MTH\$SINCOS	*****	X	01
MTH\$CEXP	*****	X	01	MTH\$SINCOSD	*****	X	01
MTH\$CLOG	*****	X	01	MTH\$SINCOSD_R5	*****	X	01
MTH\$COS	*****	X	01	MTH\$SINCOS_R5	*****	X	01
MTH\$COSD	*****	X	01	MTH\$SIND	*****	X	01
MTH\$COSD_R4	*****	X	01	MTH\$SIND_R4	*****	X	01
MTH\$COSH	*****	X	01	MTH\$SINH	*****	X	01
MTH\$COS_R4	*****	X	01	MTH\$SIN_R4	*****	X	01
MTH\$CSIN	*****	X	01	MTH\$SQRT	*****	X	01
MTH\$CSQRT	*****	X	01	MTH\$SQRT_R2	*****	X	01
MTH\$DACOS	*****	X	01	MTH\$SQRT_R3	*****	X	01
MTH\$DACOSD	*****	X	01	MTH\$STAN	*****	X	01
MTH\$DACOSD_R7	*****	X	01	MTH\$STAND	*****	X	01
MTH\$DACOS_R7	*****	X	01	MTH\$STAND_R4	*****	X	01
MTH\$DASIN	*****	X	01	MTH\$STAND_R5	*****	X	01
MTH\$DASIND	*****	X	01	MTH\$STANH	*****	X	01
MTH\$DASIND_R7	*****	X	01	MTH\$STAN_R4	*****	X	01
MTH\$DASIN_R7	*****	X	01	MTH\$STAN_R5	*****	X	01
MTH\$DATAN	*****	X	01	OTSS\$CVT_D_T_R8	*****	X	01
MTH\$DATAN2	*****	X	01	OTSS\$CVT_G_T_R8	*****	X	01
MTH\$DATAND	*****	X	01	OTSS\$CVT_H_T_R8	*****	X	01
MTH\$DATAND2	*****	X	01	OTSS\$CVT_C_TB	*****	X	01

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes														
ABS	00000000 (0.)	00 (0.)	NOPI	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE				
\$VMSS\$VECTOR	00001787 (6023.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	PAGE				

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.72
Command processing	135	00:00:00.60	00:00:03.85
Pass 1	799	00:01:03.40	00:01:50.18
Symbol table sort	0	00:00:01.57	00:00:02.25
Pass 2	475	00:00:17.00	00:00:41.89
Symbol table output	1	00:00:00.79	00:00:01.75
Psect synopsis output	0	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	1441	00:01:23.48	00:02:40.68

The working set limit was 2400 pages.
319752 bytes (625 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1092 non-local and 0 local symbols.
3082 source lines were read in Pass 1, producing 93 object records in Pass 2.
3 pages of virtual memory were used to define 2 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/LIS=LISS\$:VMSVECTOR/OBJ=OBJ\$:VMSVECTOR MSRC\$:VMSVECTOR/UPDATE=(ENH\$:VMSVECTOR)

0438

AH-BT13A-SE
VAX/VMS V4.0

**DIGITAL
CONFIDENTIAL**

EQUIPMENT
INITIAL AND

CORPORATION
PROPRIETARY